IN-18-CR
310157

P-56

A Mobile Satellite (MSAT-X) Network Definition

C. C. Wang
T.-Y. Yan

August 15, 1990

NASA

A Mobile Satellite (MSAT-X) Network Definition

C. C. Wang
T.-Y. Yan

August 15, 1990

N/\S/

ABSTRACT

This report, No. 164, describes the network architecture development of the Mobile Satellite Experiment (MSAT-X) project for the past few years. The objective of this report is twofold. First, it summarizes the results and findings of the network research activities carried out under the MSAT-X project. Second, it provides a framework upon which the mobile satellite systems (MSSs) operator can design a commercial network. With this goal in mind, a sample network configuration and its capability are included under the projected scenario.

PREFACE

In 1983, the Jet Propulsion Laboratory (JPL), as lead center for the National Aeronautics and Space Administration's (NASA's) Mobile Satellite Communications Program, was chartered to develop high-risk advanced ground technologies for future MSSs. The primary purpose of MSAT-X is to develop these technologies and validate them by conducting experiments up through the time the first-generation commercial MSSs are launched. In particular, MSAT-X network development is meant to demonstrate the feasibility of the resource-sharing technology, which, although subject to the constraints of the MSAT-X mobile environment, maximally utilizes the sparse frequency spectrum. Several alternatives for components were evaluated before the MSAT-X network was implemented. A Frequency Division Multiple Access (FDMA) architecture was selected because of the maturity of the related technologies. In 1984, an Integrated-Adaptive Mobile Access Protocol (I-AMAP) algorithm was developed for the MSAT-X; it was based on the Demand Assigned/FDMA (DA/FDMA) concept.

In 1985, a study by Signatron, Inc., concluded that the software development effort, not including the hardware deliverables, would require a substantial resource commitment by NASA/JPL. Developing such software would mean direct competition with industry in providing services, rather than technology development. Therefore, the network research activities carried on under MSAT-X have been limited to the laboratory, with the expectation that the technology can be transferred to industry if desired. The actual operational software could be jointly developed and specified by the MSSs operators.

On May 31, 1989, the Federal Communications Commission (FCC) granted the mobile satellite service license to the American Mobile Satellite Corporation (AMSC). Advanced technologies developed under MSAT-X might be transferred to AMSC. This report presents the system architecture that NASA/JPL can apply to MSSs; the architecture is based on the preliminary system characteristics filed by AMSC with the FCC. The DA/FDMA scheme is described, and the results of work on the associated network access protocol developed by JPL are presented. Both the projected system capability and the network system spectral efficiency for various traffic conditions are given in this report.

The report is organized into eleven sections. The introduction is Section I, and the MSAT-X network is reviewed in Section II. The communication-interconnection aspect of the network is discussed in Section III. The MSAT-X network structure is described in Section IV. In this structure, there are two basic protocols: One is the channel access protocol, which is discussed in Section V; the other is the link-connection protocol, in Section VI. The error-control techniques used in the MSAT-X project are discussed in Section VII. The packet structure is discussed in Section VIII. A description of two testbeds developed for experimentally simulating the channel access protocol and data-link protocol, respectively, is presented in Section IX. A sample network configuration is given in Section X. Finally, Section XI presents some future network activities of the MSAT-X project.

CONTENTS

CONTENTS (contd)

Figures

CONTENTS (contd)

SECTION I

INTRODUCTION

The concept of providing mobile communications via a geostationary satellite has been studied extensively in the past few years [1,2]. The objective is to provide the impetus for the deployment of low-cost, spectrally efficient mobile satellite services for locations and applications where terrestrial mobile systems (e.g., cellular mobile radio) are not commercially viable [3,4]. By placing a "relay tower" at a height of 22,300 miles, a Mobile Satellite System (MSS) can provide ubiquitous radio communication to vehicles roaming in remote or thinly populated areas. However, the mobile/portable nature of the terminals imposes special problems. The size and cost constraints on the terminals and antennas, together with the multipath and shadowing that are typical in a MSS channel, imply that efficient technologies that conserve resources are imperative.

In 1983, the Jet Propulsion Laboratory (JPL), as lead center for the National Aeronautics and Space Administration's (NASA's) Mobile Satellite Program, was chartered to develop the high-risk, advanced ground technologies related to the future MSSs. The primary purpose of the Mobile Satellite Experiment (MSAT-X) is to develop such technologies and validate them experimentally through the first generation of commercial mobile satellites, which are expected to be launched in 1993. The purpose of the MSAT-X network development is to demonstrate the feasibility of the resource-sharing technology that maximally utilizes the sparse natural resource of the frequency spectrum, subject to the constraints of the MSAT-X environment. The primary applications supported by the network are:

(1)   Mobile telephone (voice only)

(2)   Mobile dispatch (voice and data)

(3)   Data message transfer

In 1983, JPL evaluated several alternatives to implement the MSAT-X network. The Frequency Division Multiple Access (FDMA) architecture was selected for the following reasons:

(1)   MSAT-X was planned to be compatible with the future mobile telephone service using 4.8-Kbps digital codecs. Bandwidth separation of 5 KHz seemed to be adequate and efficient.

(2)   One of the main system considerations was to offer low-cost terminals to general subscribers. To reduce the costs borne by individual subscribers, two types of antennas would be developed. Both antennas could be mounted on vehicles and have a gain of about 3–5 dBi for an omnidirectional antenna, and 8–10 dBi for a directional antenna. This limits the bandwidth of the antenna subsystem.

(3)   MSAT-X was planned to support short data messages (of approximately 1 Kbit) and a low message-generation rate per vehicle. The single-channel per carrier (SCPC) approach was an appropriate choice.

In an FDMA architecture, the limited MSAT-X frequency band is segmented into equally sized channels. All voice, data, and control traffic will share the channels in an efficient and integrated fashion. To achieve this, the concept of the MSAT-X network is divided into two categories, namely the broadcast concept for the link from the fixed station to mobile units, and the multiple-access concept for the link from mobile units to the fixed station.

In the development of the MSAT-X network, there are four important conditions that constitute design guidelines:

(1) Each subscriber can and will communicate directly with the satellite in a single channel per carrier fashion. Some will transmit in short bursts, typical of interactive data transactions, while others will want dedicated channel assignments for telephone conversations or large file transfers. The resource-sharing algorithm must be sufficiently flexible to accommodate these nonhomogeneous requests. Voice and data modes of operation have different connection requirements and traffic statistics. All modes of operation must be integrated into a single, efficient, multiple-user access system.

(2) Subscribers to the MSAT-X network are limited to experimental users. There are requirements specific to MSAT-X that may not be necessary for the commercial MSS, or vise versa. For example, the network management center will be designed to collect statistics for future research-and-development activities related to the second- and third-generation MSSs. Billings will not be considered as a part of MSAT-X. The discussion in this report applies to the MSAT-X network; however, the concept can also apply to an operational network.

(3) In a mobile environment, the multipath fading and shadowing constantly degrade the reliability of the communications. As a result, techniques must be developed to handle the worst possible conditions. In the MSAT-X network, error-control schemes are utilized to ensure reliable communications under all conditions.

(4) The architecture of the MSAT-X network must accommodate equipment changes and changes in the network protocols. This architecture allows the infusion of new technologies as they emerge. (The construction of the JPL network testbed follows this concept.)

In considering the key multiple-access concept for the FDMA architecture, the long propagation delay between the earth and the geostationary satellite precludes the use of carrier sense-type multiple access. In order to efficiently and effectively utilize the resources, the Demand Assigned Multiple Access (DAMA) approach was considered for the network implementation. In particular, in 1984, an Integrated-Adaptive Mobile Access Protocol (I-AMAP) algorithm was developed for the MSAT-X that was based on the Demand Assigned/FDMA (DA/FDMA) concept [5]. The algorithm operates to optimize the performance between two conflicting goals:

(1) Maximize the number of subscribers that can effectively access the available bandwidth allocation.

(2) Minimize the delay between a subscriber's request for channel access and the completion of services.

The MSAT-X network uses a centralized control approach. It is managed by a network management center (NMC) that monitors and controls the network operations. In addition to the NMC, other physical elements of the MSAT-X network are mobile terminals (MTs) and base stations (BSs), as shown in Fig. 1-1. Also shown in the figure is a gateway (GW), which is used to interface the network with a public switched telephone network, though the GW is not considered in the MSAT-X network structure.

Although the NMC is the brain of MSAT-X, the successful execution of the DAMA algorithm requires close interactions among various software protocol handlers residing in different network elements. The DAMA algorithm includes detailed signaling specifications related to connection establishment and relinquishment, information transfer, error control, error recovery, and subscriber log-on and log-off procedures among various elements. Since the proposed DAMA scheme is relatively new, a substantial portion of the design effort should be devoted to the software engineering. In 1985, a
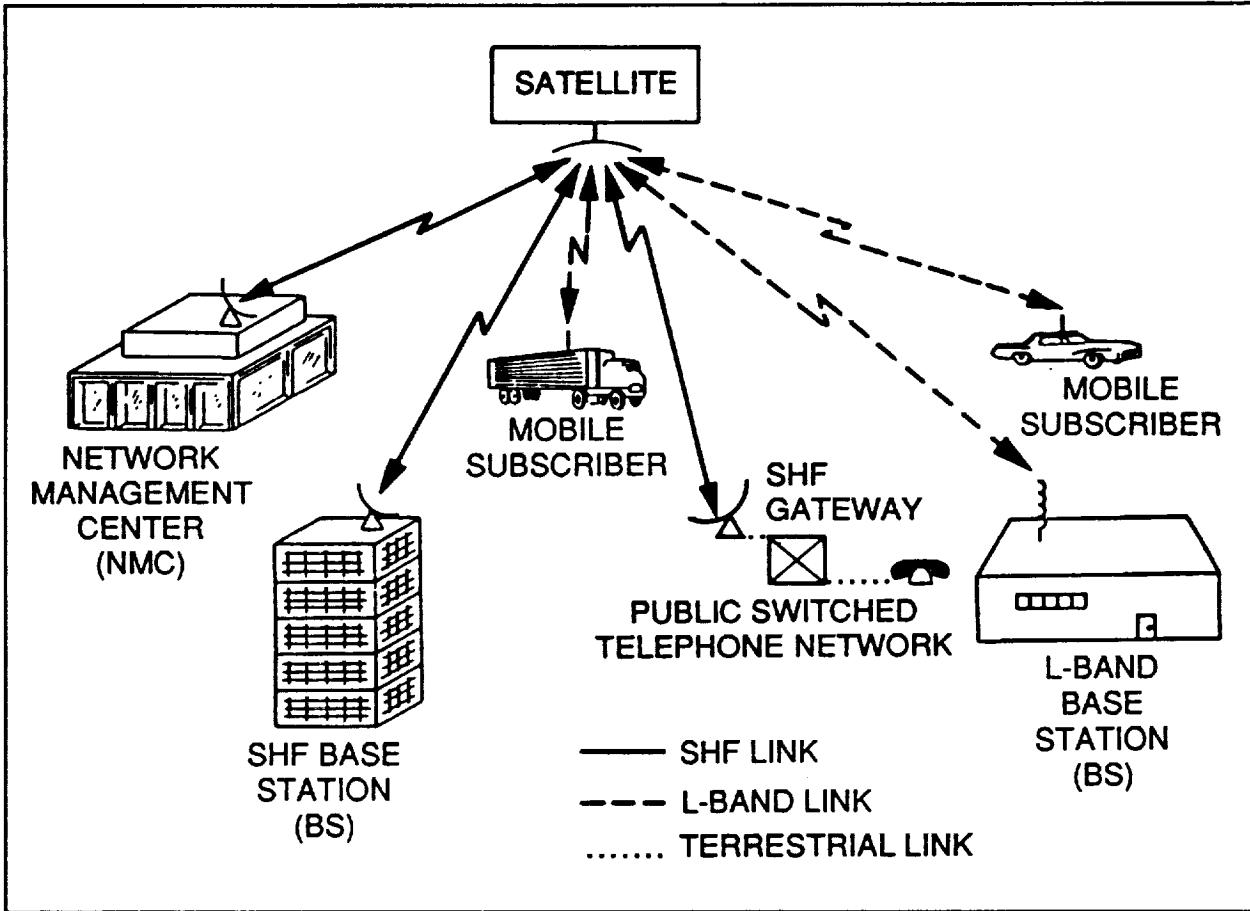
Figure 1-1. Elements of the MSAT-X Network

study by Signatron, Inc., concluded that the software-development effort, not including the hardware deliverables, would require a substantial resource commitment by NASA/JPL [6]. Developing such software would compete directly with industry to provide services rather than technology development. Therefore, MSAT-X's research activities have been limited to the laboratory, with the expectation that the technology could be transferred later to industry. The actual operational software could be jointly developed with, and specified by, the MSS operator.

On May 31, 1989, the U.S. Federal Communications Commission (FCC) granted a mobile-satellite service license to the American Mobile Satellite Corporation (AMSC). Advanced technologies developed under MSAT-X might be transferred to AMSC. This report presents the system architecture applicable to a MSS by NASA/JPL, based on the preliminary system characteristics filed by AMSC with the FCC. The DA/FDMA scheme is described, and the results of the associated network-access protocol developed by JPL are presented. Both the projected system capability and the network system spectral efficiency for various traffic conditions are given in this report.

In the DA approach, each subscriber must initiate a request (for service), then be instructed to start the service during a certain time window. A channel-access scheme must be adopted for subscribers to initiate requests. A random access scheme, such as pure or slotted ALOHA, is the simplest scheme to implement. Various modifications of the basic ALOHA channel were developed to improve the system performance. However, even with the modifications, the ALOHA channel is unstable and does not present good throughput at high traffic levels. Collision-resolution algorithms can improve the throughput at the expense of higher complexity. Various channel-access techniques have been investigated. Due to the long propagation delay in the MSAT-X environment, a modified slotted

ALOHA and a free-access tree algorithm have been developed for making requests. The trade-off between these two schemes will be discussed in this report.

Once the request is accepted by the NMC, a connection protocol will be executed between the two communication endpoints. An automatic retransmission request (ARQ) error-control technique will be used to ensure reliable communications under the mobile fading condition. The ARQ technique requires an error-detection code and proper packetization of the message. However, an uncoded ARQ system is very inefficient because of the large number of retransmissions, which results in an unacceptably long message delay. The efficiency of an ARQ system can be increased by using a forward error correction (FEC) code to improve the communication link performance. The FEC code must be selected appropriately to maintain an adequate data throughput.

In the ARQ technique, the shorter the packet, the more the overhead, since each individual packet requires a fixed amount of overhead. A shorter packet length implies an increase of the total transmission time. On the other hand, a longer packet length results in a larger packet error rate (PER), meaning that more retransmissions are required, which decreases the throughput. In 1986, the optimization of the packet length and other specifications of the packet structure due to the use of the error-correcting code was studied. The results are presented in this report.

The interactions among the ARQ scheme, the trellis code, block interleaving, and packet replication make theoretical analysis intractable for mobile fading channels with memory. Simulation efforts were initiated in 1988 to evaluate the system performance, and two testbed simulators were constructed. Detailed descriptions of these two testbeds are discussed, respectively, in later sections. Using the two simulators, one can experimentally study the delay-throughput performance for various protocols in mobile fading environments. The testbeds were designed so that, if the protocol is changed, the structure can be changed easily simply by replacing either the subsystem hardware or the software, as necessary.

The objective of this report is twofold. First, it summarizes the results and findings of the network research activities under the MSAT-X project for the past few years. Second, it provides a framework upon which the MSS operator can design a commercial network. Toward this goal, a sample network configuration and its capability are included under the projected scenario in this report. The MSAT-X network is reviewed in Section II. The communication-interconnection aspect of the network is discussed in Section III. The MSAT-X network structure is described in Section IV. In this structure, there are two basic protocols: One is the channel-access protocol, which is discussed in Section V, the other is the link-connection protocol in Section VI. The error-control techniques used in the MSAT-X are discussed in Section VII. The packet structure is discussed in Section VIII. A description of two testbeds developed for experimentally simulating the channel-access protocol and link-connection protocol, respectively, appears in Section IX. A sample network configuration is given in Section X. Finally, Section XI presents some future network activities of the MSAT-X project.

SECTION II

THE MSAT-X NETWORK

A.   NETWORK ELEMENTS

The basic network elements of the MSAT-X are the mobile terminals (MTs), the base stations (BSs), and the network management center (NMC). Functionally, the MTs and the BSs serve as ultimate connection endpoints; that is, they are the direct sources and sinks of subscriber traffic. The last network element mentioned above, the NMC, is responsible for the allocation of the (satellite) channel resources and for enabling the connections among the network subscribers. It also monitors and controls the network operations. Only network-control information is passed to and from the NMC.

There are two types of mobile terminals, namely, mobile telephone terminals and mobile dispatch terminals. Mobile telephone terminals have only a voice-generation/reception capability. Base stations and mobile dispatch terminals have both a voice- and data-generation/reception capability. The NMC exclusively uses data signaling for all of its control communications, although a voice-reception capability will exist for the purpose of monitoring voice-channel communication characteristics.

B.   PHYSICAL TOPOLOGY

Figure 2-1 illustrates the physical topology of the MSAT-X network. The satellite provides only a repeater (with an appropriate frequency shift) function. The satellite is assumed to have a multiple L-band beam capability, as indicated by L_1 through L_N, with a single full-coverage Ku-band (super high frequency, SHF) beam. Distinct Ku-band channels correspond to the L-band in each beam. The MTs use L-band to communicate with the satellite; their mobility between beams makes the physical topology dynamic. Each L-band channel is translated to/from a unique Ku-band channel for MT communications with the NMC or a BS (or a GW for the MSS). Communications between the NMC and a BS (or a GW) will take place via a Ku-band cross-strap through the satellite but could also take place via terrestrial links. All elements of the network must be able to communicate with the NMC in a single hop through the satellite. MT-to-MT communications will require two hops through the satellite, with a suitably configured BS providing the necessary Ku-band translations.

C.   LOGICAL TOPOLOGY

The logical connectivity of the network is defined as the matrix of all permissible communications among network elements, i.e., who is authorized to communicate with whom. The logical connectivity depends upon both mode of operation (dispatch, telephone, etc.) and the association of particular elements as a subnetwork (e.g., Ace Trucking Company). Table 2-1 shows the logical connectivity matrix for the MSAT-X. For example, a mobile telephone terminal only communicates with its affiliated BSs. Enforcement of this connectivity is accomplished as part of the channel-assignment function of the NMC.

Each MT (either telephone or dispatch terminal) may be affiliated with a particular BS (for the MSAT-X) or group of BSs (for the MSS) and may only communicate with that BS or that group of BSs. (Voice communication among MTs is not considered in the MSAT-X.) Dispatching is normally organized along company or fleet lines; thus, members of a dispatch fleet would be allowed only to communicate with others in the same fleet. Within a dispatch fleet there will be terminals with the capability of voice, data, or voice and data, and thus, the logical connectivity must be defined so that the terminals wishing to communicate are compatible. Communications between dispatch terminals are routed through the same affiliated BS or the originating terminal's BS (if they are not affiliated with the
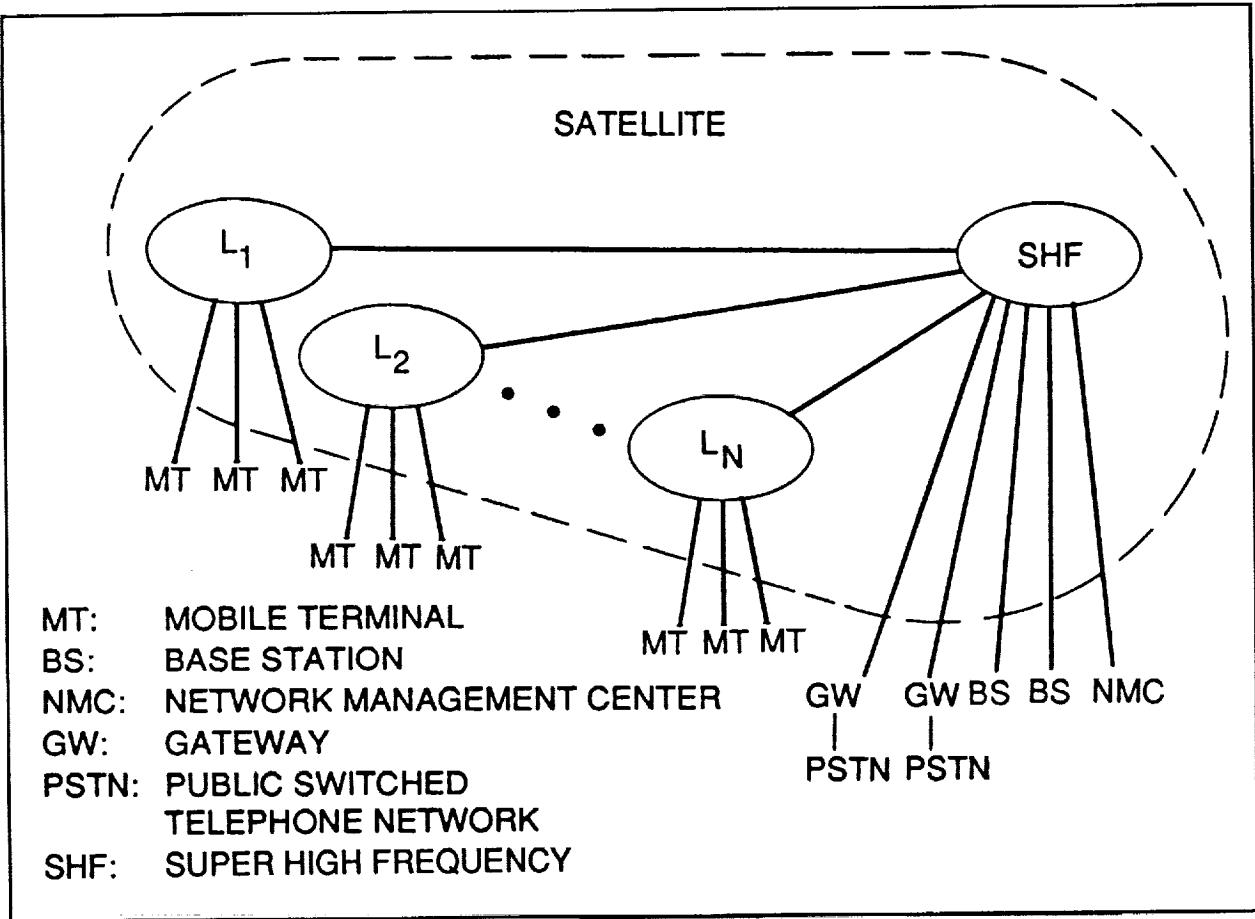
Figure 2-1. Physical Topology of the MSS Network

same BS). This connection requires two hops between the earth and the satellite. The setup of this structure is for the connecting BS to be able to monitor the connection and perform other supervising services, such as billing and emergency interruption. All MTs and BSs can communicate with the NMC itself for the purpose of establishing connections.

Table 2-1. Logical Connectivity Matrix of the MSAT-X

DESTINED PARTY / ORIGINATING PARTY	MODES OF EXPERIMENTS		MTs SUBSCRIBED TO THE SAME BSs	MTs SUBSCRIBED TO DIFFERENT BSs	BASE STATION
MOBILE TERMINAL (TELEPHONE AND DISPATCH)	VOICE	TELEPHONE	NO		YES
		DISPATCH	DOUBLE HOP THROUGH THE SAME BS	DOUBLE HOP THROUGH THE ORIGINATED PARTY BS (POSSIBLY CROSS CHARGED TO DESTINED PARTY BS)	(BCQ) YES
	DATA	DISPATCH			YES
		LONG FILE	NO		YES
BASE STATION	VOICE	TELEPHONE	NO		
		DISPATCH	YES		YES (THROUGH Ku-BAND CROSS STRAP)
	DATA	DISPATCH			
		LONG FILE	YES	NO	

2-3

SECTION III

COMMUNICATION INTERCONNECTION CHARACTERISTICS

A. BASIC CONNECTION MODES

A variety of connections are required to support the MSAT-X services. From the point of view of network capability, it is natural to partition the types of connections into either closed-end or open-end connections. These two basic connection modes are identified as follows:

(1) In a "closed-end connection," the channel-holding time is predetermined based upon the amount of data to be transferred. Normally, a closed-end connection is used for transmitting short messages between the MT and the BS, or the MT and another MT. Furthermore, although the message must be delivered in a timely fashion, the service of this kind of connection need not be in real time. This permits the use of ARQ error-control techniques to improve the fidelity of the message.

(2) In an "open-end connection," the duration of the service is not specified at the time the request is initiated. Normally, an open-end connection is used for a voice conversation or long data-file transfer. Voice connections occur in real time, so that a fixed delay equal to the propagation time occurs between a transmission and its delivery to the end user. This real-time requirement limits the application of ARQ error-control techniques to improve the link transmission. However, real time is not required for transferring a long data file. Hence, the ARQ error-control schemes can be utilized to improve the reliability.

B. CHARACTERISTICS OF VARIOUS CONNECTION MODES

Data transmission for closed-end connections will be performed from MT to BS, BS (dispatch) to MT(s), or MT to MT. Connections will follow a queuing discipline, whereby pending data transfers are scheduled by the NMC at the time of their request. Point-to-point data transfers will include acknowledgments. Dispatch to many mobiles, i.e., "multicast," will not include an acknowledgment.

Voice conversation and open-end file transfer will be performed between the MTs and BSs only. (The MT-to-MT open-end connection is not required for the MSAT-X.) Connections are made based on the availability of the channel and the destination. The connections will be kept as long as the subscriber wishes. Below is the description of different connection characteristics.

1. Short Message Transmission

Short message transmission using a closed-end connection will consist of two one-way data connections separated in time. One is the forward link for transmitting the message from a mobile unit to its destination, i.e., another mobile unit or a BS; the other is the return link for conveying the acknowledgment from the destination to the sender. The transmitting data message is queued for service and can be divided into multiple packets (the number depends on message length). A study by Signatron, in 1985, under a contract with JPL, concluded that, in order to efficiently utilize the channel throughput, the number of packets contained within the same closed-end data message should be limited to 16 [6]. (For a message that requires more than 16 packets, an open-end connection will be used.) A unique acknowledgment will be required for the entire message transmission. The corrupted packets within the same message will form an additional message and make a separate transmission attempt.

3-1

2.   Long File Transfer

The long data-file transfer for the open-end connection will be provided by a full-duplex link. The file will be packetized. Any packet corrupted because of channel errors will be retransmitted during the same connection session. That is, the channel will not be relinquished until all packets within the same file are correctly received.

3.   Voice Dispatch

The voice dispatch provided by the open-end connection will be half duplex, connecting the dispatcher (at the BS) to one or more MTs on a single beam or connecting an MT to one or more MTs all within the same beam. With the exception of a mobile-originated call to a dispatch operator, all connections will be made under a block-calls-cleared (BCC) protocol, meaning that the call will be returned with a busy tone if there is no circuit available or the destination is engaged in another service. The mobile-to-dispatch calls will be processed under a blocked-calls-queued (BCQ) protocol because of the possibility of many mobile users desiring access to a single dispatcher. A BCQ protocol for this specific connection will prevent repetitively blocked channel requests from fleet mobiles when the dispatcher is already engaged in another connection.

4.   Telephone

Telephone connections will be full-duplex voice from/to MT to/from its affiliated BS. Connections will be made on a BCC basis.

SECTION IV

NETWORK STRUCTURES

A. DYNAMIC CHANNELIZATION

In the MSAT-X network, MTs follow a DA approach to access the network. In such an approach, the fundamental resource, i.e. the satellite and its operational frequency bands, is shared by all the network entities. Each MT must make a connection request to the NMC before the actual connection is established. The entire L-band within an antenna beam is divided into 5-kHz channels. At any given instant, the channels within the same beam can be grouped according to the traffic of each service that they currently support. There are four functional types of channels: request channels, command channels, open-end channels, and closed-end channels. Request channels are used for MTs to access the NMC for initialization and making connection requests. Command channels are used by the NMC to provide control information to the network users regarding acknowledgments of request-channel traffic, connection attempts by other users, and network status information. One command channel is designated as the "wake-up" channel to provide information for terminals to log onto the network. Open- and closed-end channels are used, respectively, for the actual transfer of user traffic in the course of an open- and a closed-end connection. The distinction between open- and closed-end channels corresponds to the two connection types described in Section III.

Efficient utilization of the channels is accomplished by means of the integrated-adaptive mobile access protocol (I-AMAP) algorithm [5]. The channels are dynamically partitioned into the functional groups described above, based on the demand of open- or closed-end connections. Figure 4-1 illustrates an example of channel partition; the channels supporting the same function need not be contiguous but are shown as such for ease of presentation.
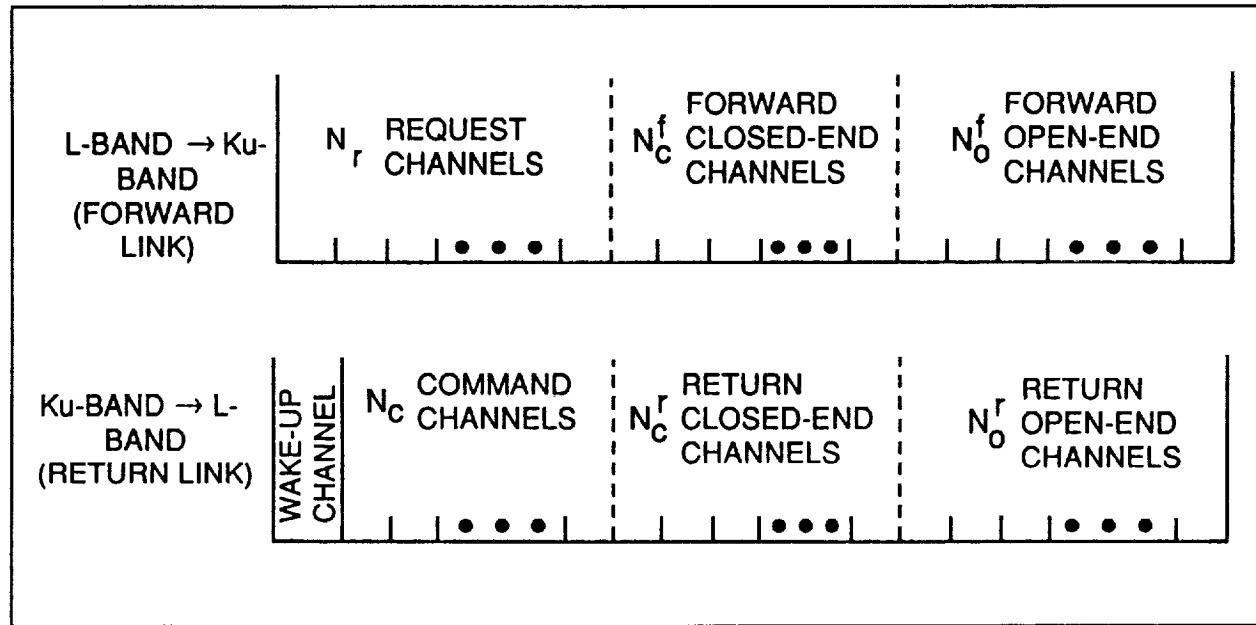

Figure 4-1. An Example of Channel Partition of the MSAT-X

B. INTEGRATED-ADAPTIVE MOBILE ACCESS PROTOCOL ALGORITHM

The available bandwidth of the MSAT-X is configured into paired channels, one in the forward direction (from L-band to Ku-band), and one in the backward direction (from Ku-band to L-band). In either link, the bandwidth is grouped into three types of channels, as indicated in Fig. 4-1 (the wake-up channel in the return link is considered to be one of the command channels). A message sent in a particular channel will be acknowledged in the corresponding paired channel. The NMC is responsible for the assignment of channels. The NMC periodically estimates the arrival rate for open- and closed-end connections and finds the optimal partition among three types of channels in the forward direction (see Fig. 4-1) to minimize the overall end-to-end closed-end message delay under a constraint of blocking probability for open-end connections. (The paired channels in the return link will be automatically partitioned accordingly.) Each channel may be assigned as any type of channel by the NMC. The NMC must inform the MTs of all channel reassignments. To ensure that a MT will not be accessing an obsolete request channel, reassignments will only be activated after a certain waiting period, typically one round-trip propagation delay. The identities of the request channels are transmitted on the wake-up channel. Communications between the MTs and the BS, or between the MTs and the NMC, will be routed through the satellite. Communications between the BS and the NMC are assumed to be routed over terrestrial links or Ku-band cross-strap links.

When a MT first logs onto the network, it tunes to the wake-up channel, finds the identities of the currently assigned request channels, and sends a registration message on one of the request channels. This registration message will be transmitted and acknowledged as if it were a connection request.

When the MT is not engaged in an ongoing connection, it is under the control of the NMC. Whenever it wishes to make a connection, it monitors the wake-up channel and finds the identities of the currently assigned request channels. It then gains access to the NMC by transmitting a connection request using a channel access protocol (CAP) over a set of selected request channels. Any collision in the request channel(s) must be resolved, based on the CAP implemented, to ensure the error-free reception of the request message at the NMC. (A detailed description of the CAPs will be given in Section V.) The connection request consists of the origin and destination addresses and, whether it is an open-end or a closed-end request. In addition, in the case of a closed-end data request, the MT also divides the message into fixed-length packets and saves them in its transmitting buffer. This process will be discussed in detail in Section VII. The packets are sequentially numbered so that they can be acknowledged individually. The length (in packets) of the message being sent will also be included in the request packet. New messages that are generated before all packets in the transmitting buffer are properly acknowledged will be stored in the queuing buffer and then transferred to the transmitting buffer when that becomes available. The request for connecting the message will be made until it is transferred to the transmitting buffer. BSs make connection requests over either SHF–SHF cross-strapped links or possibly via terrestrial links with the NMC.

After sending out a request, the requesting MT then waits for a channel assignment from the NMC at the paired command channel. If an assignment is not received within a preset time-out period, or if the assignment message contains errors (as determined by a cyclic redundancy check, CRC), the requesting MT will retransmit the request according to the CAP. In this procedure of making requests, the NMC is responsible for processing all requests for communications and accordingly assigns channels for connection; in the case of a closed-end connection, this also includes scheduling transmissions, which will be discussed in Section VII.

For an open-end connection, when the connection request arrives successfully at the NMC, the NMC checks whether the destination party (either BS or MT) is busy in other service or whether all open-end channels are occupied. In either case, the NMC sends a busy status to the requester. Otherwise, the NMC assigns the request to one of the available open-end channels and sends an assignment message to both the requester and the destination party. (Assignment messages are treated as the acknowledgment to the request messages.) The assignment message contains the identity (ID) of the

requester and destination, and the ID of the assigned open-end channels. After receiving the assignment message, the requester tunes to the assigned open-end channel and starts the conversation or transmits an open-end file. At this stage, control responsibility is transferred to the intermediate BS from the NMC. The cognizant BS monitors the connection and signals the NMC in the event of connection failure or disruption. The channel is held as long as the service lasts. Note that the BS must be a party involved in an open-end connection (see Section III). When the conversation or file transfer is completed, the BS will initiate the relinquishment procedure by sending a packet directly to the NMC for taking down the channel. This packet contains IDs of both parties and the ID of the channel that is to be taken down. Once the connection is relinquished, control responsibility returns to the NMC. Figure 4-2 shows the procedure for making an open-end connection. In the case of an open-end file transfer, the file is partitioned into packets. Each packet is error-detected, and the packet is negatively acknowledged individually. (The channel for sending acknowledgments is the same as that used for transmitting, since in this case, a full-duplex channel is used.) The packets that are received with errors will be negatively acknowledged and then retransmitted. The retransmission is performed during the same connection period. That is, the connection will not be terminated until all packets within the same file are positively acknowledged.
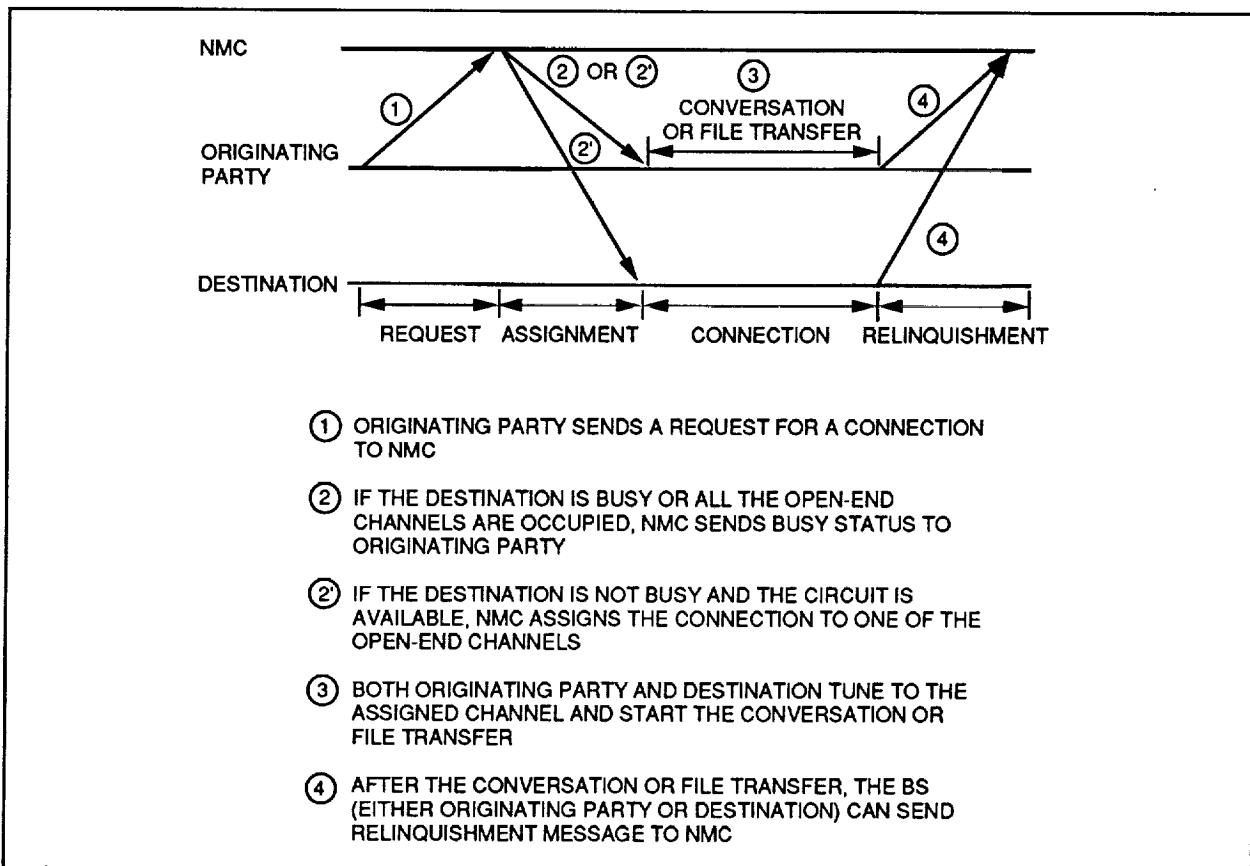
Figure 4-2. Procedure for Making an Open-End Connection

For a closed-end connection, when the connection request arrives successfully at the NMC, the NMC assigns, on a first-come-first-serve basis, a closed-end channel and its paired acknowledgment channel for this particular connection. The closed-end channel selected is that with the least amount of backlog. The NMC then sends channel-assignment messages to the requester. The assignment message contains IDs of both the requester and destination, the ID of the assigned closed-end channel, a holding time d_H,

and the ID of the acknowledgment channel. The holding time is calculated so that all messages previously assigned to the same closed-end channel for transmission will be completely sent out. The same assignment message is also sent to the destination. At that moment, the NMC hands the control to the intermediate BS. After receiving the assignment message, the requester waits for the specified holding time d_H and sends over the designated closed-end channel all the packets in its buffer that correspond to this message. The requester then waits for an acknowledgment at the paired acknowledgment channel. During the connection period, the cognizant BS monitors the service and then returns the control back to the NMC, if the service is successfully accomplished. Figure 4-3 shows the procedure for making a closed-end connection. In any event of service disruption, the sender will not receive the acknowledgment message within a preset time-out period (the propagation delay to the destination and back, plus the transmission and processing delay of the message), and then the entire procedure will be restarted from the very beginning.
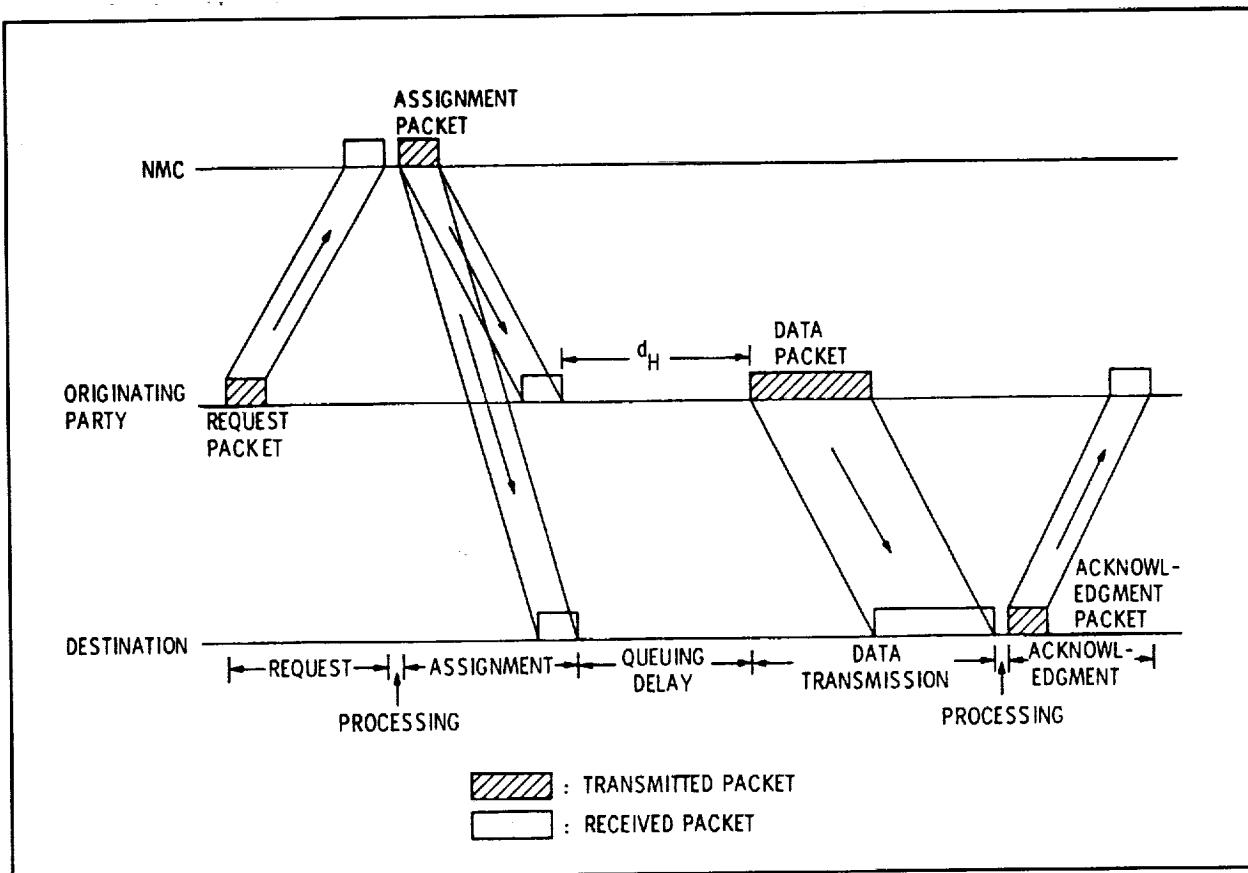
Figure 4-3. Procedure for Making a Closed-End Connection

As will be discussed in Section VII, the message in the closed-end connection is also packetized. The acknowledgment message received will indicate those packets that are received with errors. In case the packets are not acknowledged, the sender will attempt a retransmission. The packets in its transmission buffer that are positively acknowledged will first be discarded. Then, a connection request will be made on behalf of all unacknowledged packets. This procedure is repeated until all packets in its buffer that correspond to the message have been positively acknowledged.

SECTION V

CHANNEL ACCESS PROTOCOL (CAP)

The procedure for sending connection requests and waiting for their acknowledgment is the CAP. Due to the long propagation delay in the MSAT-X environment, a modified slotted ALOHA scheme with a maximum throughput of 0.368 packet per slot per channel was recommended in 1984. The low throughput of this CAP is a bottleneck for the entire MSAT-X system throughput, since channel resources must be allocated to carry the nonrevenue-generating connection request traffic. In addition, the modified slotted ALOHA protocol is inherently unstable. Simulations show that this type of protocol must be operated at a 25- to 30-percent throughput backoff from the maximum capacity. This implies that the maximum usable throughput rate for the request channels is at most 0.28 packet per slot per channel. Furthermore, in practice, the slotted ALOHA protocol is operated at less than 0.15 packet per slot per channel to ensure system stability.

In random access communications, collisions occur when two or more packets are simultaneously transmitted over a channel. Collision resolution algorithms are methods that govern the retransmission of collided packets so that every collided packet is eventually successfully transmitted with finite delay, and all transmitters involved become aware of exactly when this occurs. A packet collision is said to be resolved precisely at the point when all transmitters involved simultaneously become aware that the colliding packets have been successfully transmitted. These algorithms operate in a slotted-time environment and require broadcast feedback information from the receiver to all the transmitters of the channel collision state in each slot. The algorithms can improve the system throughput at the expense of slightly higher complexity. For this consideration, in 1987, JPL developed a new CAP, namely the free-access tree algorithm, for making connection requests in the mobile satellite network [7]. This new CAP is inherently stable and dead-lock free at throughput up to 0.402 packet per channel per slot for a network with 3 request channels. The trade-off between the two schemes is given in this section.

A.   MODIFIED SLOTTED ALOHA

Slotted ALOHA is a random access scheme. In the slotted ALOHA system, all MTs are synchronized for each slot time, which equals one request-packet length plus a guard time. This guard time is required to accommodate the propagation-delay difference between different ends of a beam coverage. Upon receiving a request from the subscriber, the MT waits until the beginning of the most immediate slot time and sends a request packet over a randomly selected request channel. Then, the MT waits for an acknowledgment from the NMC. If the acknowledgment is not received within a time-out period, which usually equals a round-trip delay plus the processing time, the MT retransmits the same request after a back-off delay. This back-off delay is uniformly distributed between 0 and K slots. If a negative acknowledgment is received, meaning that the request packet was received (without collision) in error, the MT will immediately retransmit the packet without a back-off delay. The reason for using the back-off delay is to alleviate any further collisions during the retransmission among the packets that previously collided. Note that the request channels in the MSAT-X are multiple parallel slotted ALOHA channels, which are much more efficient than the traditional single slotted ALOHA channel that is usually used in the packet radio system. An analysis of delay-throughput performance for this multichannel slotted ALOHA system was conducted in 1985 at JPL [8]. The results indicated that the additional amount of request traffic that the network can support by adding each request channel is equal to the amount of request traffic that can be supported by one channel. Furthermore, the stability of a multichannel slotted ALOHA system increases with the increasing number of channels. Consider an ALOHA system with a back-off delay at 10 slots and an operating point at 0.3 packet per slot per channel. According to [8], the so-called first exit time (FET), the average time until all packets are blocked after reaching the specified operating point, of a 2-channel ALOHA system is 25 times that of a

1-channel ALOHA system. When the number of ALOHA channels increases to 3, the FET becomes 100 times that of the 1-channel ALOHA system.

A replication technique was also investigated over the basic mechanism of the slotted ALOHA protocol at JPL, in 1986, to improve the performance in a fading environment [9]. With this technique, the packet is contiguously repeated a fixed number of times at each transmission attempt. The technique has been employed in the call-origination signaling procedure for cellular telephone systems [10] to combat fading and increase message reliability. The delay-throughput performance for the slotted ALOHA protocol with packet replications in the presence of fading was analyzed in [9]. It has been shown that greater throughput and smaller delay may be achieved by using repetitions. Numerical results show that, for a moderately noisy channel with a packet error rate (PER) of, for example, 30 percent, two consecutive transmissions of a packet are preferable to a single transmission, in terms of both the packet delay and the channel throughput. However, in the case of a very small PER environment, using packet replication will degrade the delay performance because of the increase in packet size (meaning the increase in packet collisions.) In MSAT-X, the PER is estimated to be between 10 and 15 percent. Although repeating the connection-request packet may reduce the delay for making connection requests, the channel throughput will be reduced.

Note that the use of a smaller back-off delay K can result in a higher possibility of further collision with or without replication. On the other hand, a large value of K can cause unnecessary delay and degrade the throughput. In 1987, a modification to the basic slotted ALOHA was developed at JPL to improve the delay-throughput performance. (The Signatron study also recommended this modification to the slotted ALOHA). The modified slotted ALOHA protocol utilizes a so-called exponential back-off delay that presents a trade-off between the above-described two extremes of large K and small K. That is, when a packet collides for the first time, the MT will use a smaller value of K (e.g., 10) for the first attempt at retransmission. Then, the MT will double the value of K every time the same packet collides during the retransmission. The delay-throughput performance of this modified slotted ALOHA multiple access scheme will be shown in the next subsection, when the performance of two CAPs are compared. Although the packet replication is not used in the comparison, it can be predicted that the performance will be further improved in a noisy environment by sending repeated packets.

B.   FREE-ACCESS TREE ALGORITHM

The free-access tree algorithm is also a random access scheme. It operates on multiple request channels. Let N be the total number of request channels available for MTs to access the network. Like the modified slotted ALOHA CAP, these N channels are also synchronously slotted. Packet transmissions are allowed only at the start of a slot. The fundamental operation of this algorithm can be specified as follows:

(1)   Newly arriving packets are immediately transmitted at the first available time slot on a randomly selected channel.

(2)   Any collision over one of the N channels is resolved on a channel-by-channel basis. That is, the system will not start to resolve any collision that occurred on channel i until all collisions on channels 1 through i-1 are resolved.

(3)   During the process of resolving a collision that occurred on channel i, any retransmitted packet can be sent over any one of the N channels.

In this scheme, every MT must be able to receive feedback information from the NMC about whether packet collisions occur in any of the N channels during every single slot (even when the user itself is not transmitting a packet). At any given time, each MT is allowed to have, at most, one packet either ready for transmission or in the process of being transmitted. Any new packet that arrives before the

previous colliding one is resolved will be stored in the buffer. After the previous packet is successfully received, only the earliest packet in the buffer will be released. Each terminal also implements a counting algorithm to determine the slot time at which it must retransmit the packet. Assume that channels are indexed by 1, 2, ..., N, and that the tree algorithm always starts to resolve collisions from the channel of the smallest number to that of the largest number. Let C be the value of the counter in a user of interest. Packet transmissions are initiated at the start of a slot only when C = 0. The counter value C is incremented or decremented at the end of each slot, based on the channel-state feedback information of all N parallel channels. The rules for this counting algorithm are described as follows:

(1)   C is initially set to 0.

(2)   C must be 0 when any new packet starts to be transmitted. If C = 0 at the start of a slot, then C remains unchanged if either the user terminal does not have a packet ready for transmission or a packet has been sent without collision.

(3)   When C = 0 at the start of a slot and the user's transmitted packet has just suffered a collision on channel j in that slot, then C is set to be i, where i is the number of channels with an index smaller than j that has packet collisions in the same time slot.

(4)   When C > 0 at the start of a slot, C will be decreased by 1 if there are no packet collisions over all N channels. If packet collisions occur in k out of N channels in that slot, C will be increased by k-1.

(5)   If the packet has not been successfully transmitted, it will be retransmitted immediately following the slot in which C reaches 0.

Figure 5-1 shows the flowchart of the counting algorithm that starts at the beginning of each time slot.

It is clear that the value C is positive if and only if it has a collided packet waiting for retransmission. The collided packet's next retransmission time occurs when C becomes zero. Since this is a relatively new CAP, the following example will illustrate the operation of this protocol. Figure 5-2 illustrates the evolution of the packet transmissions in this example. Suppose that initially in slot 1, users A, B, and C select channel 1; users D and E select channel 2; and users F and G select channel 3 for transmitting packets. Also, users H and I generate new packets in slot 1. Figure 5-3 illustrates the evolution of the counter value C of user F. So all three channels suffer a collision during slot 1. First, the collision among packets from users A, B, and C must be resolved, then the collision among those from users D and E, and finally the collision among those from users F and G. Note that in Fig. 5-3, the value C of user F's counter is equal to 2 at the end of slot 1 because of the collisions in channels 1 and 2. So in slot 2, the new packets from users H and I are transmitted along with the first retransmissions of the collided packets of users A, B, and C. Suppose that in slot 2, users A, B, and I select channel 1 and users C and H select channel 2. Also, a new packet is generated from user J in slot 2. User F's counter value C is now incremented by 1 to 3 at the end of slot 2, since there are two channels with collisions in slot 2. Since the collision among packets of users A, B, and I in slot 2 must first be resolved before the collision among those of users C and H in slot 2, packets from users A, B, I, and J are next transmitted in slot 3. It follows from Fig. 5-2 that the collision among packets from users A, B, and I in slot 2 is finally resolved at the end of slot 5. So in slot 6, the resolution of the collision among packets from users C and H in slot 2 is started. Figure 5-2 shows that this is completed at the end of slot 8. At this time, the initial collision among users A, B, and C in slot 1 has also been resolved. So in slot 9, the resolution of the initial collision among users D and E in slot 1 is started and resolved. Hence, in slot 10, the resolution of the initial collision among users F and G in slot 1 is started. Note in Fig. 5-3 that user F's counter value C first reaches 0 at the end of slot 9, thus indicating its first retransmission in slot 10. This collision is finally resolved at the end of slot 11. In this example, 12 packets are transmitted using 11 slots for a per channel throughput of 0.364.
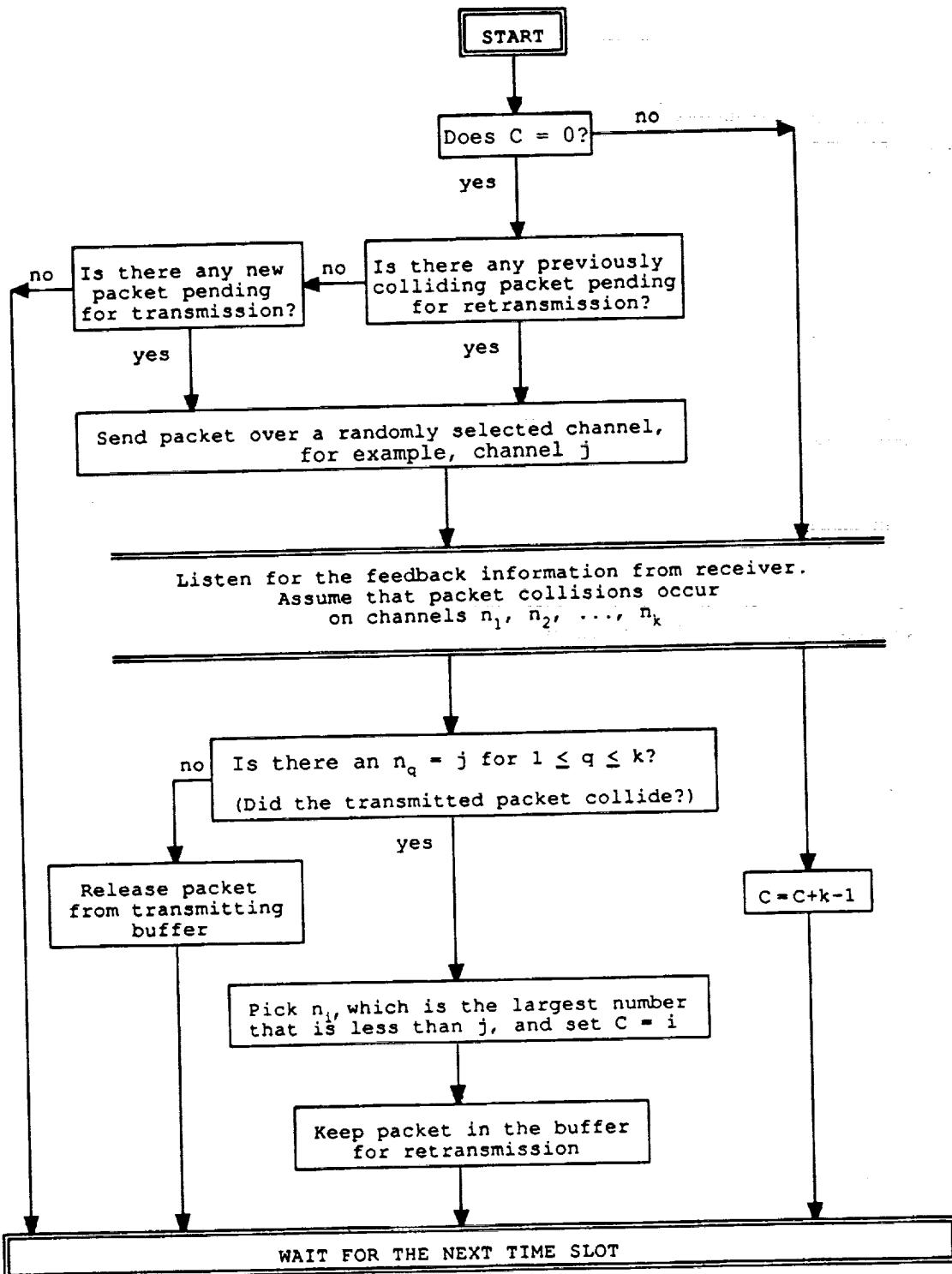
Figure 5-1. Flowchart of the Counting Algorithm in the Free-Access Tree Algorithm

SLOT #	1	2	3	4	5	6	7	8	9	10	11
CHANNEL 1	A, B, C	A, B, I	A, I	A	B	C, K	–	–	D	–	–
CHANNEL 2	D, E	C, H	B, J	–	J	–	C	H	–	F, G	F
CHANNEL 3	F, G	–	–	I	–	H, L	K	L	E	–	G
NEW PACKETS	H, I	J	–	–	K, L	–	–	–	–	–	–

Figure 5-2. Packet Arrivals and Transmissions

SLOT #	1	2	3	4	5	6	7	8	9	10	11
COUNTER VALUE	2	3	4	3	2	3	2	1	0	0	0

Figure 5-3. Counter Value C of User F at the End of Each Slot

Note that the above-described multiple-channel free-access tree algorithm assumes that the feedback information for each slot is known to the users immediately at the end of the prevailing slot. Of course, this assumption is valid only when the round-trip propagation delay is negligible, as compared with the packet transmission time. (In any case, the round-trip propagation delay can be included in the slot time in addition to the packet transmission time, since the throughput degradation is negligible.) However, in the case of significant round-trip propagation delay, a number of interleaved tree algorithms can be operated. Specifically, suppose that the round-trip propagation delay between transmitters and receivers is equal to d-1 slots. Then, d interleaved copies of the previously described multiple-channel free-access tree algorithms can be operated. The first algorithm operates in slots 0, d, 2d, ... ; the second algorithm operates in slots 1, d+1, 2d+1, ... ; and so on. The implementation advantage of free-access tree algorithms over other collision resolution algorithms, such as blocked-access algorithms, can be substantial for large propagation delays.

The free-access tree algorithm described above has been analyzed and implemented in software in 1988 at JPL. Simulated performance results for a different N from 2 to 6 were obtained for Poisson infinite-user population statistics. These simulations are consistent with the theoretical analysis. Stable system behavior was observed. Figure 5-4 shows the delay-throughput performance of the free-access tree algorithm. In this figure, the round-trip propagation delay is assumed to be zero, without loss of generality.
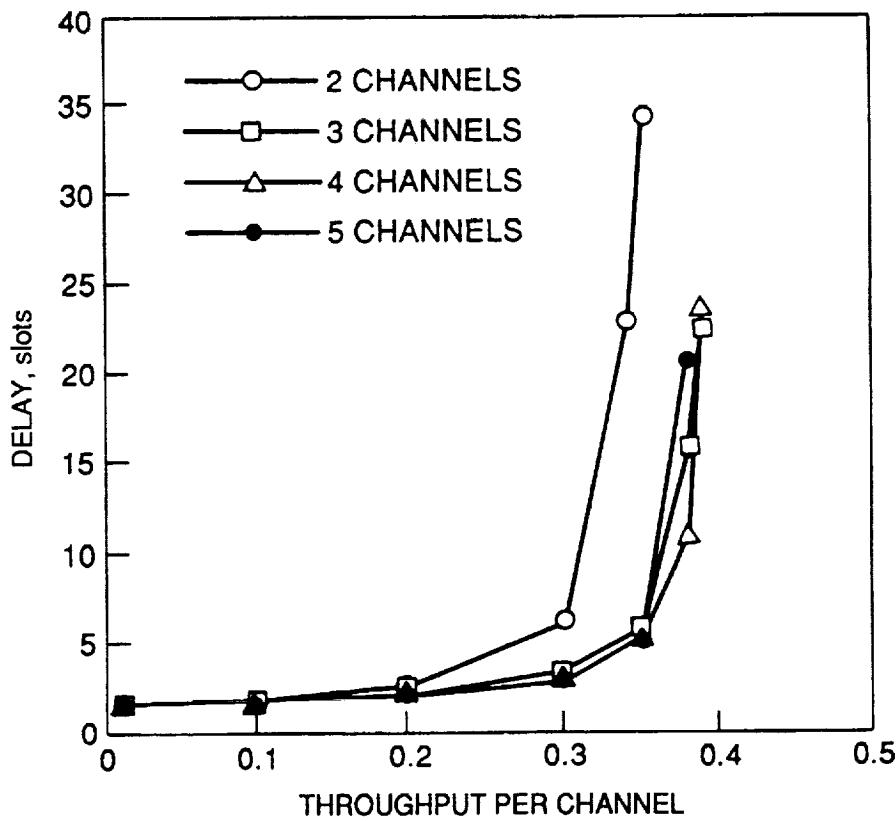

Figure 5-4. Simulation Results of the Free-Access Tree Algorithm

This free-access tree algorithm has been compared with the modified slotted ALOHA algorithm. In the modified slotted ALOHA algorithm, an initial maximum back-off delay for retransmission is set at 15 slots, and this value is doubled after successive collisions of a transmitted packet. Figure 5-5 shows the comparison between the modified slotted ALOHA and free-access tree algorithm for 3 request channels. It can be seen that the tree algorithm has a delay-throughput characteristic that is uniformly superior to the modified slotted ALOHA protocol. The delay-throughput characteristic is also extremely flat over a wide range of traffic input rates for the tree algorithm, thus indicating a very robust behavior, even under traffic loads close to capacity. It can be seen in Fig. 5-5 that the free-access tree algorithm presents a significantly improved throughput as compared with the modified slotted ALOHA.
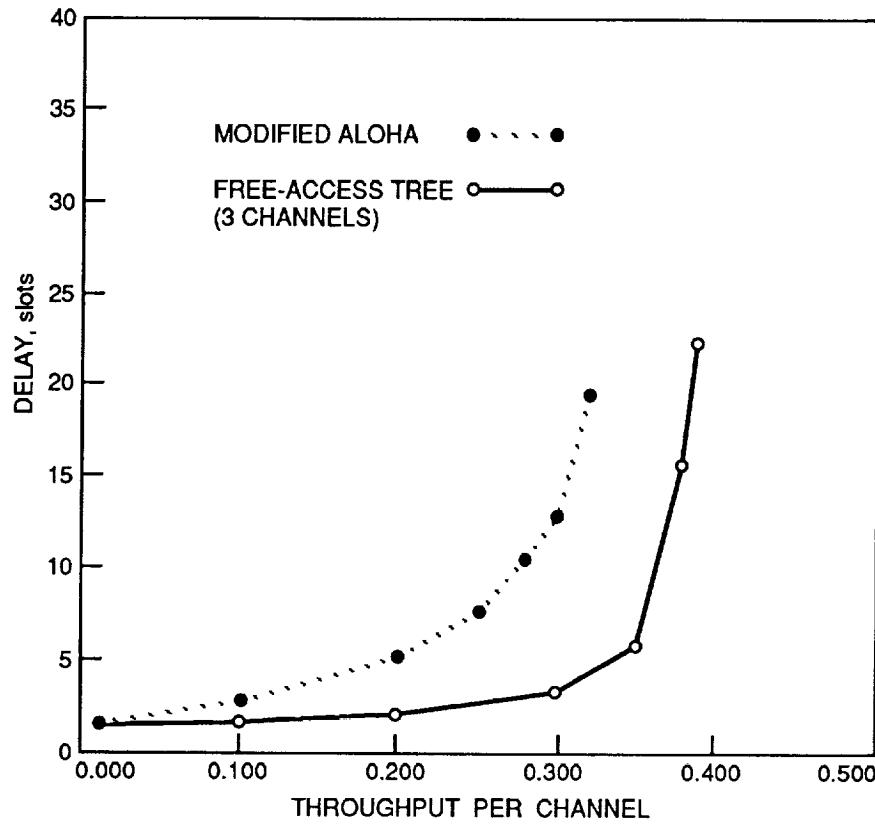
Figure 5-5. Performance Comparisons Between Two CAPs

In conclusion, the free-access tree algorithm shows superior delay-throughput performance in comparison with the slotted ALOHA. It also has a uniformly higher maximum throughput than the modified slotted ALOHA protocol. Moreover, system behavior is absolutely stable up to these maximum throughput rates, as opposed to the inherently unstable ALOHA protocols. However, at low traffic levels, the improvement is not significant. Note that the complexity of the tree algorithm is higher than that of the modified slotted ALOHA. Considering the trade-off between the performance and complexity, the modified slotted ALOHA scheme is recommended for a system with low traffic (e.g., MSAT-X) because of its simplicity. Nevertheless, for a network of moderate and heavy traffic (e.g., MSS), the free-access tree algorithm is recommended for achieving high throughput and better stability.

SECTION VI

CONNECTION PROTOCOL

In the conventional DA/FDMA scheme, the subscriber who requests a channel can use it in any type of connection (including packetized data, voice, etc.) for himself or among subordinate subscribers, until the channel is relinquished. The subscriber will be billed for the duration of using the channel. This scheme is particularly effective for circuit-oriented connections because of the relatively long circuit-holding time.

It is well known that circuit-oriented connections are inefficient for packet communications. In the I-AMAP algorithm, the pool of channels maintained by the NMC can be adaptively used as request, command, open-end, or closed-end channels. Packet-oriented connections are communicated over closed-end channels, while circuit-oriented connections are communicated over open-end channels. The NMC adaptively adjusts the partitions between the various channel types to guarantee the level of services provided for each type of connection. Typically, short message transmissions use closed-end connections, while voice conversations or large file transfers use open-end connections.

A.   OPEN-END CONNECTION

The open-end connection setup and clearing procedures resemble the corresponding procedures in the telephone system. To establish an open-end session, the calling party (either MT or BS) sends a CALL-REQUEST packet to the NMC through one of the request channels, following the CAP. Any collision or channel-corrupted packet will be resolved by the CAP. After successfully receiving the CALL-REQUEST packet, the NMC sends an INCOMING-CALL packet to the called party if the called party is not busy and an open-end channel is available (referred to as the line-not-busy case, see Fig. 6-1); otherwise, it sends a CLEAR-INDICATION packet back to the calling party (referred to as the line-busy case, see Fig. 6-2). The INCOMING-CALL packet contains the ID of the assigned open-end channel. This is the assignment packet from the NMC to the called party, as stated in Section IV. After successfully receiving the INCOMING-CALL packet, the called party responds with a CALL-ACCEPTED packet to the NMC through the assigned open-end channel. The CALL-ACCEPTED packet can be viewed as an acknowledgment to the INCOMING-CALL packet.

After successfully receiving the CALL-ACCEPTED packet from the called party, the NMC sends a CALL-CONNECTED packet to the calling party. If the CALL-CONNECTED packet is lost or is not received correctly, it is considered to be a negative acknowledgment to the CALL-REQUEST packet, and the calling party will retransmit the CALL-REQUEST packet according to the CAP. The CALL-CONNECTED packet contains the ID of the assigned open-end channel. This is the assignment packet from the NMC to the calling party, as stated in Section IV. After successfully receiving the CALL-CONNECTED packet, the open-end connection is set up and the calling party can start its conversation or file transfer over the assigned open-end channel.

Note that the BS must be a party involved in an open-end connection (see Section III). After an open-end connection is successfully accomplished (in the case of file transfer, it means that all packets are positively acknowledged), the BS initiates the link-clearing procedure by sending a CLEAR-REQUEST packet to the NMC through either Super High Frequency−Super High Frequency (SHF−SHF) cross-strapped or terrestrial links. The link-clearing procedure is dual to the line-not-busy case of the link-setup procedure with the CALL-REQUEST, INCOMING-CALL, CALL-ACCEPTED and CALL-CONNECTED packets being replaced by the CLEAR-REQUEST, CLEAR-INDICATION, DATA TERMINAL EQUIPMENT (DTE)-CLEAR-CONFIRMATION and DATA COMMUNICATION
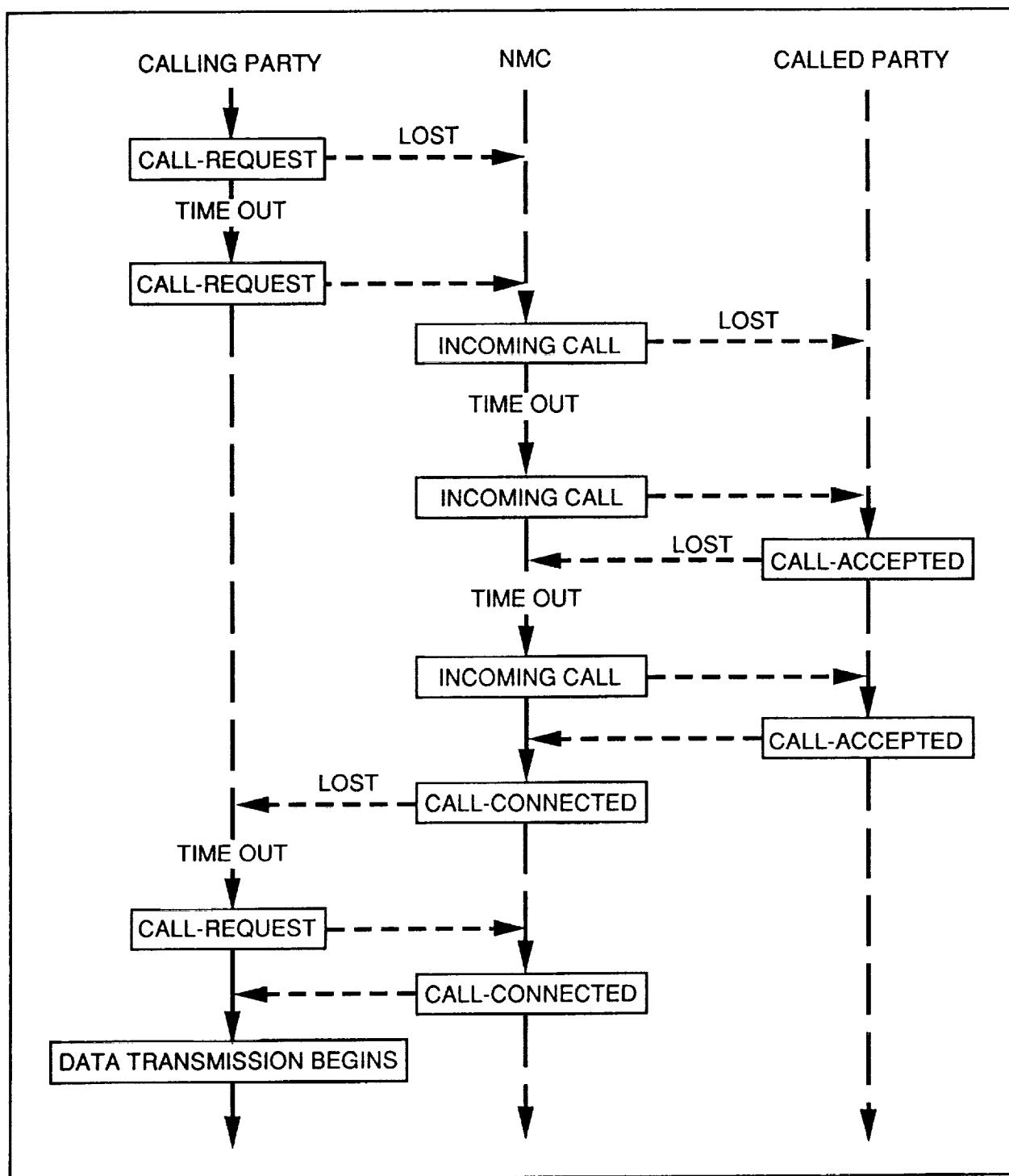
Figure 6-1. Open-End Connection Procedure (Line-Not-Busy Case)
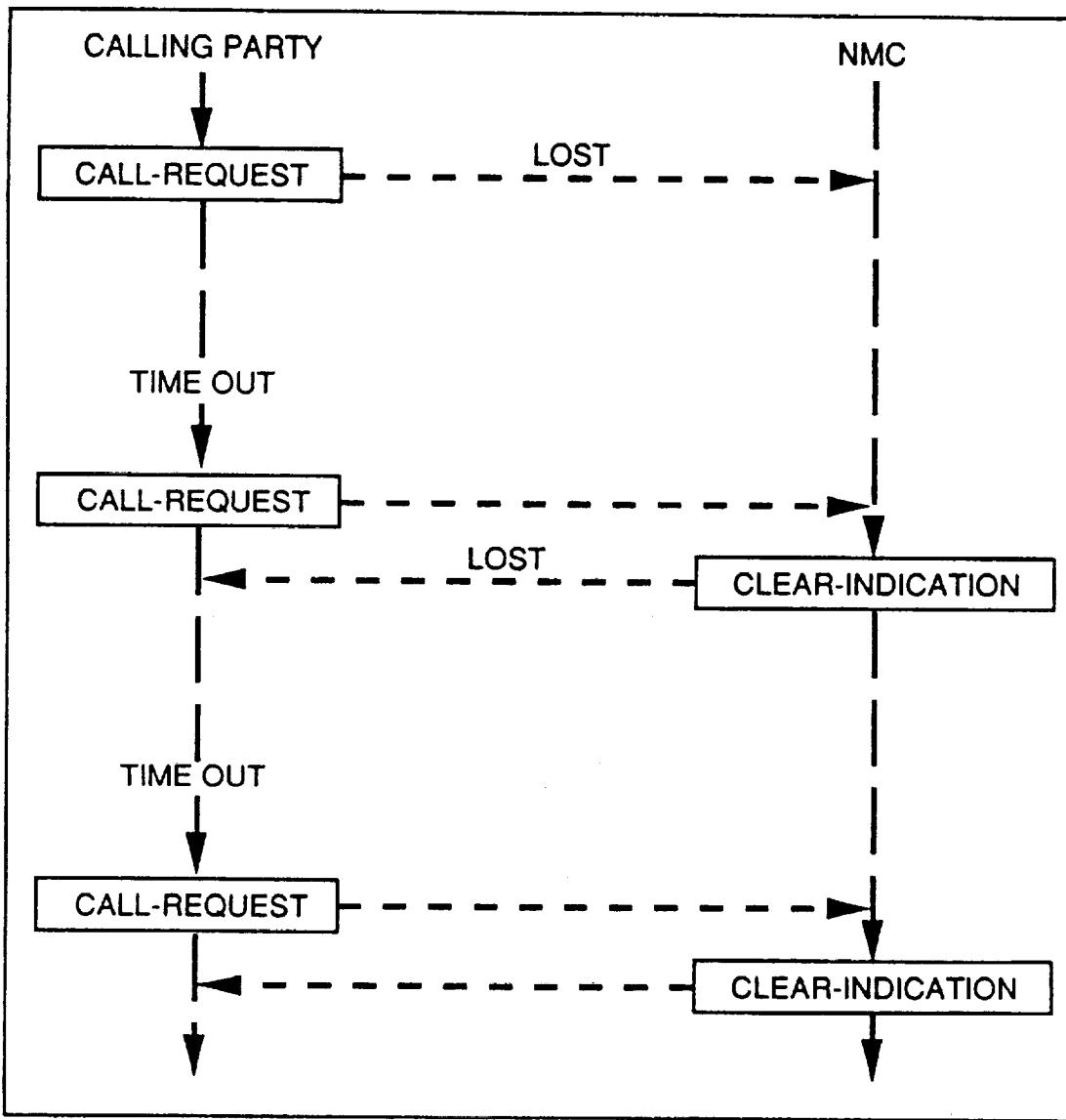
Figure 6-2. Open-End Connection Procedure (Line-Busy Case)

EQUIPMENT (DCE)-CLEAR-CONFIRMATION packets, respectively. This procedure for taking down an open-end connection is described in Fig. 6-3.

B.   CLOSED-END CONNECTION

The closed-end connection is similar to the open-end connection, except that the NMC does not need to send the INCOMING-CALL packet to the called party, and the called party does not need to respond with the CALL-ACCEPTED packet. To establish a closed-end connection, the calling party sends a CALL-REQUEST packet to the NMC through one of the request channels, following the CAP. Any collision or channel-corrupted packet will be resolved by the CAP.
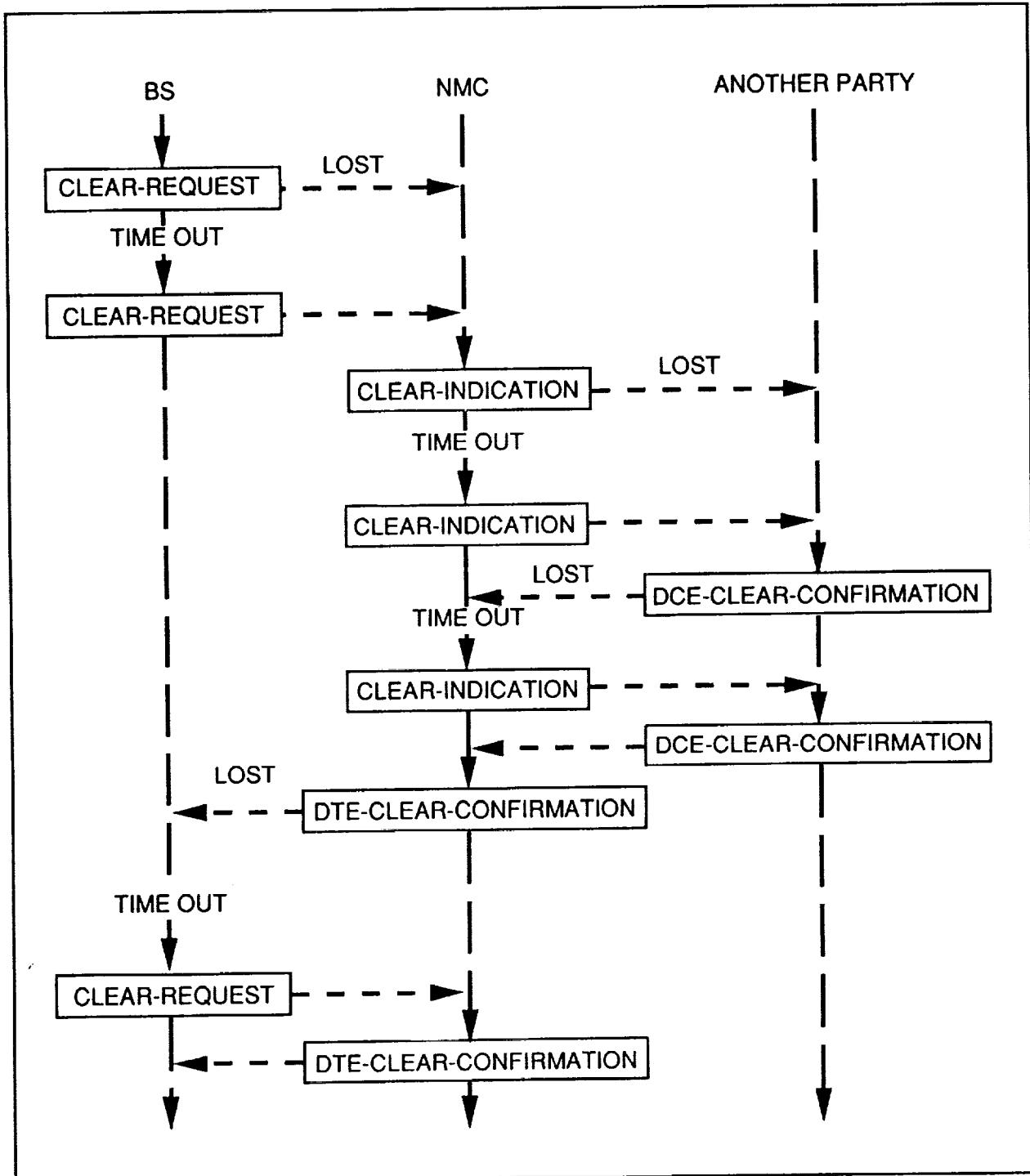
Figure 6-3. Procedure for Taking Down an Open-End Connection

After successfully receiving the CALL-REQUEST packet, the NMC will assign this particular request to one of the available closed-end channels with the shortest backlog, on a first-come-first-served basis. The NMC broadcasts a CALL-CONNECTED packet, which is the assignment packet described in Section IV, to the calling and called parties. If the CALL-CONNECTED packet is lost or is not received correctly, it is considered to be a negative acknowledgment to the CALL-REQUEST packet, and the calling party will retransmit the CALL-REQUEST packet according to the CAP. The CALL-

CONNECTED packet contains the scheduled time window and the identity of the assigned closed-end channels. The CALL-CONNECTED packet also serves as an acknowledgment to the CALL-REQUEST packet.

After successfully receiving the CALL-CONNECTED packet from the NMC, the closed-end connection is set up. Note that the time and duration of the connection is known to both the calling and called parties. The calling party waits until the scheduled transmission time, then sends the DATA packet on the assigned closed-end channel. Then it waits for an ACKNOWLEDGMENT packet from the called party at the scheduled time window. If any portion of the procedure is not successfully accomplished, the calling party will retransmit the CALL-REQUEST packet, and the procedure starts from the very beginning.

In closed-end connections, both the channel-occupation time and the transmission-starting time are available to the NMC, calling parties, and called parties for closed-end connections. Therefore, no clearing procedure is necessary.

ERROR CONTROL

Error-control techniques are used to ensure reliable transmissions in the mobile fading environment. The technique depends on the type of connection for message services. In the MSAT-X, messages are segmented into packets. Closed-end connections are used for short messages, typically less than 16 packets per message. Open-end connections will be more efficient to use for messages (or files) longer than 16 packets.

A.   SELECTIVE REPEAT ARQ FOR CLOSED-END DATA TRANSFER
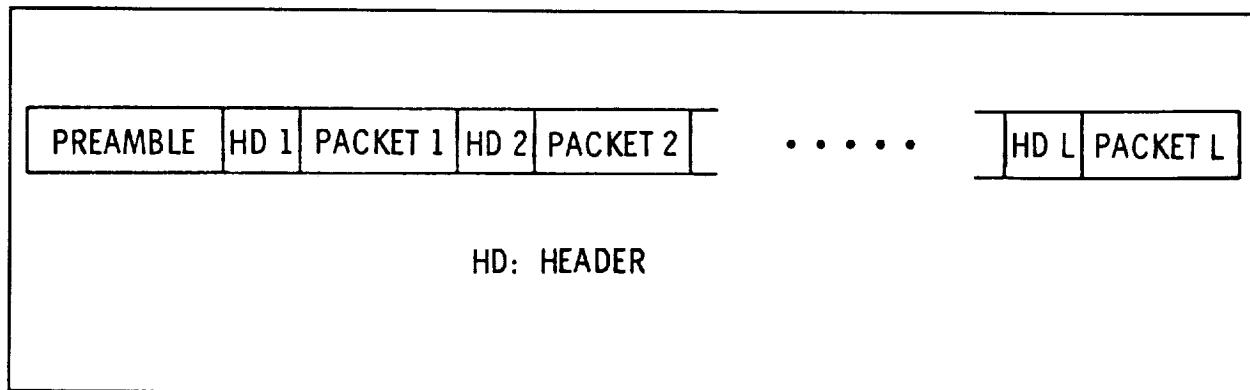
Figure 7-1.   Transmitted Data Message

In the closed-end connection, after the message is segmented into fixed-length packets, CRC bits are computed and appended to each packet for error detection. A header, which includes the packet sequence number, is also added to each packet. Packets are sent continuously (without gaps), as shown in Fig. 7-1. Prior to transmitting the message, a preamble is attached to the front of the packet stream for the receiver to acquire. The receiver checks for any transmission errors in each packet by calculating the CRC sequence. After receiving the entire message, the receiver will include the packet sequence numbers of those packets received in error in the ACKNOWLEDGMENT packet (refer to this particular packet as a negative acknowledgment, or NAK) and send this NAK back to the transmitter through the assigned acknowledgment channel. Then, the transmitter will make a separate connection request on behalf of the NAK'ed packets. A separate assignment-and-acknowledgment cycle will be required for each retransmission. This procedure continues until the receiver correctly receives every single packet within the message, and an ACKNOWLEDGMENT packet (referred to as the positive acknowledgment packet, or ACK) indicating this status is received by the transmitter. In case the NAK or ACK is not received by the transmitter, the entire, previously transmitted message will be retransmitted. Figure 7-2 illustrates this file-transfer scheme for the closed-end connection. In the MSAT-X, due to the importance of the acknowledgement packet, all NAK or ACK packets are repeated twice (see subsection E).

B.   SELECTIVE REPEAT ARQ FOR OPEN-END FILE TRANSFER

In the open-end connection, the message (or file) is also divided into packets. Within each packet, the CRC bits and the packet sequence number bits are also attached. As shown in Fig. 7-1, packets of the same file are transmitted continuously, with a preamble attached to the front of the entire file. At
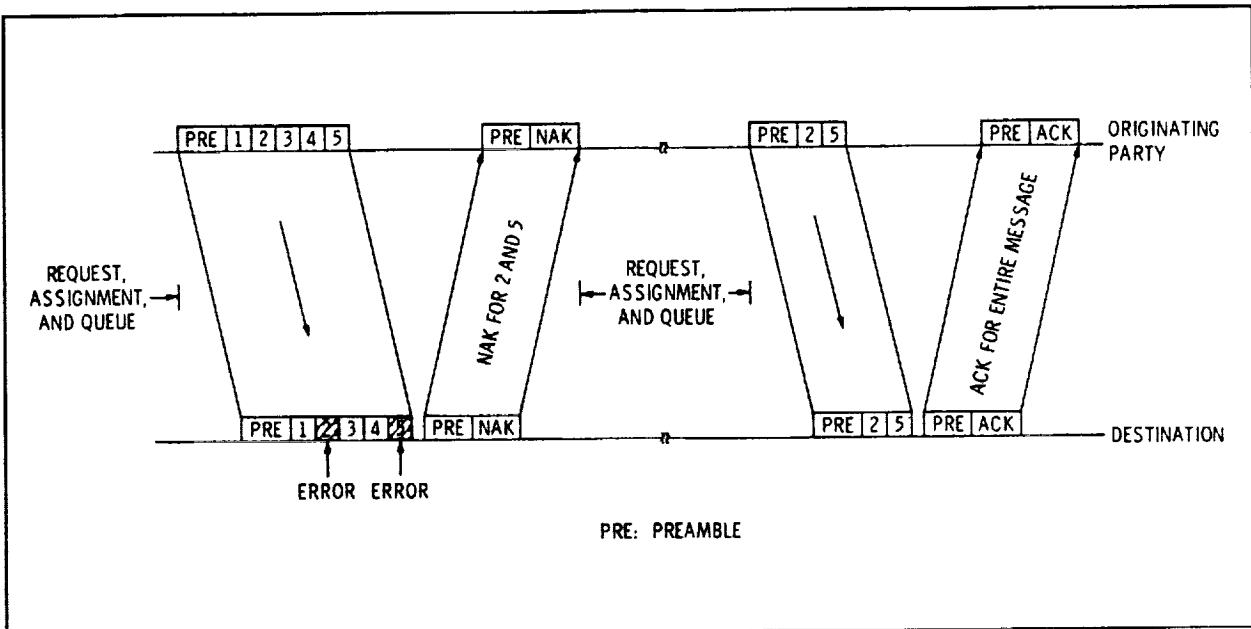
Figure 7-2. File Transfer Scheme for the Closed-End Connection

the receiver, errors are detected in each individual packet. Then, an ACK or NAK for each packet is sent back to the transmitter through the full-duplex open-end channel. Whenever the transmitter receives a NAK, or does not receive any acknowledgment (in case the acknowledgment is lost during the transmission) for a particular packet, it will stop transmitting remaining packets and resend the NAK'ed or un-ACK'ed packet. (For simplicity, in the rest of this report, the NAK always includes the un-ACK unless stated otherwise.) After resending the NAK'ed packet (or packets, if another NAK is received while resending the NAK'ed packet), the transmitter continues transmitting remaining packets within the same message. Finally, an end-of-file (EOF) packet will be sent to signify the completion of the file. This EOF packet will also be acknowledged and will be retransmitted if it is NAK'ed. When all packets (including the EOF packet) within the file are ACK'ed, a link-relinquish procedure, which was described in Section VI, starts. Figure 7-3 shows the selective repeat ARQ file transfer for this connection.

C.   FEC CHANNEL CODE

In a mobile fading environment, the short-term channel error rate can be as high as 10^{-2}. An uncoded ARQ system becomes very inefficient, due to the large number of retransmissions, which results in an unacceptably long message delay. The efficiency of an ARQ system can be increased by using an FEC code to improve the communication-link performance.

In the MSAT-X, a trellis-coded modulation (TCM) is used for both data and voice services (since voice signal is digitized) [11]. This particular TCM consists of a rate 2/3, 16-state trellis code followed by an 8-ary differential phase-shift keying (DPSK) modulator. This coding structure can provide not only an acceptable BER in mobile fading environments, but also a code rate of 2, which makes possible transmission at a 4800-bps data rate (for both data and digitized voice).

D.   SYMBOL INTERLEAVING

A Viterbi decoder is used for decoding the demodulated TCM signal. It is well known that the Viterbi decoder is powerful enough to correct the random errors but is vulnerable to burst errors. However, the mobile fading environment can easily create a long burst of errors that can dramatically

7-2
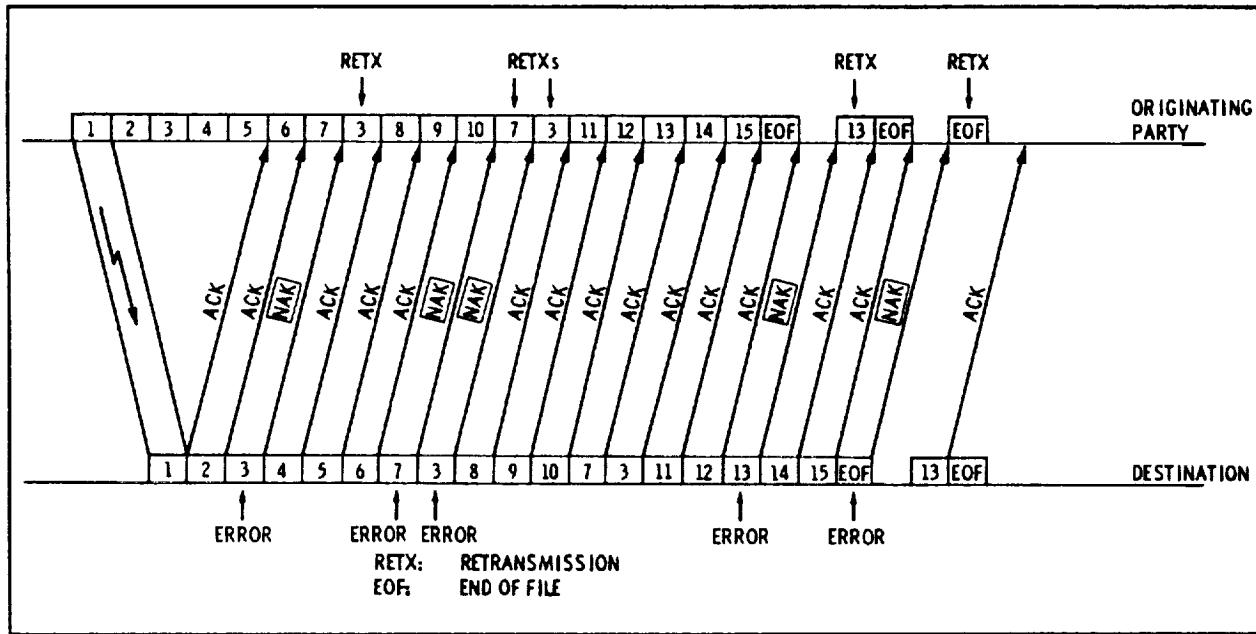
Figure 7-3. File Transfer Scheme for the Open-End Connection

degrade the performance of the TCM. Interleaving presents a solution to this problem. A 16 (depth) symbol by 8 (span) symbol block interleaver will be used behind the trellis encoder for the MSAT-X [11]. At the receiving end, before entering the Viterbi decoder, the signal is first block deinterleaved, which breaks any burst errors into scattered single errors. This makes the TCM much more robust in fading environments.

E.   PACKET REPETITION

The efficiency of an ARQ system relies on the PER after channel decoding and symbol interleaving. Packet repetition is a simple and effective coding technique to improve the PER. In this technique, the transmission is considered to be successful if any one copy is received without error. Therefore, packet repetition reduces the number of possible retransmissions and hence reduces the message delay. On the other hand, replication directly increases the packet transmission time, which results in longer end-to-end delay. The optimum value of the number of repetitions depends on the channel condition.

In the request channels, packet delay is dominated by collisions among packets. Repeating request packets will increase the probability of collision and may not be beneficial under MSAT-X environments. (According to [9], it may be beneficial when the PER exceeds 30 percent. However, this is not the case with the MSAT-X.) In addition, repeating long data packets will increase transmission time and queuing time, and then decrease the data throughput, though it may improve the PER. Therefore, it is preferable to repeat only the assignment packets and acknowledgment packets. In [12] for the MSAT-X, 2 is the optimum replication number for these two types of packets.

7-3

SECTION VIII

MESSAGE STRUCTURE

The content and the length of the message depends on the information the message will convey. Messages for log-on, request, assignment, and acknowledgment purposes carry limited information, which can be represented by one packet of fixed length. However, the length of the data message is usually long and unknown a priori. For message services using the ARQ error-control technique, data messages are divided into packets.

There are two kinds of information that are required in all types of packets. They are the packet-type indicator and the cyclic redundancy check (CRC) sequence. The packet-type indicator identifies the type of the packet and can be represented by 4 bits. The CRC sequence occupies 16 bits to detect possible packet errors. In addition, since MSAT-X uses a rate 2/3, 16-state trellis code for encoding finite length packets, 8 dummy zero bits are inserted at the end of each packet. These dummy bits force the encoded data sequence to merge to a known all-zero state at the end of each data sequence in the trellis diagram. This is critical to ensure that the received sequence can be decoded starting from an all-zero state. The other additional bits required for the different packets are described as follows.

Log-on Packet

The log-on packet is used for a MT to log onto the MSAT-X network when it enters into a new antenna beam. Its subscriber ID must be included in the packet. For the number of subscribers to be accommodated in the network, 16 bits are sufficient to identify all the subscribers. Including the packet type, CRC, dummy zero sequence, and other control information, a length of 64 bits is anticipated for the log-on packet.

Request Packet (CALL-REQUEST, CLEAR-REQUEST)

The request packet described here is for either making a connection (CALL-REQUEST) or for taking down a channel (CLEAR-REQUEST). The packet is sent by a MT or a BS to the NMC. The requester ID and the destination ID are required in such a packet. In the case of requesting a closed-end data transmission, the time required for transmission (in terms of number of packets) is also included. Therefore, a maximum of 128 bits is used for request packets.

Assignment Packet (INCOMING-CALL, CALL-CONNECTED)

Assignment packets are sent by the NMC to inform the transmitting and destination parties (MT or BS) that a connection is expected. In each of these packets, 16 bits are used to represent each transmitting party and 16 bits are used to represent the destination party; 16 bits are used to represent the channel ID where the connection will occur; 4 bits are used to represent the service type (either open-end or closed-end), and either data or voice. In the case of a closed-end connection, an additional 16 bits are used to indicate the holding time; 8 bits are used to indicate the service time (in terms of number of packets); 16 bits are used to indicate the acknowledgment-channel ID; and 4 bits are used to indicate the expected acknowledgment time (in terms of the number of round-trip delays between satellite and earth). Therefore, a length of 128 bits is sufficient for each assignment packet.

Data Packet (DATA)

As stated in Section VII, using the ARQ scheme requires packetization of the message. If the packet size is long, the probability of receiving an erroneous packet is large. Hence, more retransmissions are required to transfer the file, which results in longer end-to-end delay and lower throughput. On the other hand, shorter packet size necessitates more total overhead within the file, since each individual packet needs a fixed overhead that includes the CRC bits and other bits for representing the packet-sequence number. This implies that a longer packet transmission time is required for each packet. Consequently, there must exist an optimum packet size so that the average end-to-end delay can be minimized.

A study to find the optimum data-packet length was conducted at JPL in 1986 [12]. It concluded that the optimum size of the data packet is 256 bits for the closed-end file and 512 bits for the open-end file. However, for both cases, the study also indicated that using any packet size in the range between 256 and 512 bits shows no significant degradation of delay performance.

Acknowledgment Packet (ACKNOWLEDGMENT)

The acknowledgment packet is sent by the destination party to inform the transmitting party of the correctness of the received data packets. In this packet, 16 bits are used to represent each of the IDs of the transmitting and receiving parties. During the open-end file transfer, each transmitted packet will be individually acknowledged. Therefore, only the prevailing packet-sequence number and a simple indicator of the correctness of this packet are required. However, for the closed-end data transmission, the acknowledgment packet must correspond to the entire message, which consists of multiple packets. As stated in section IV, the maximum number of packets allowed in a closed-end data message is 16. Therefore, in the acknowledgment packet, a minimum of 4 bits must be used to indicate the number of packets in the prevailing message that were received in error. Four bits can be used to represent the packet-sequence number of each packet that was received in error, resulting in a maximum of 64 bits. Hence, a length of 128 bits is required for the acknowledgment packet.

Other Control Packets (CALL-ACCEPTED, CLEAR-INDICATION, DTE-CLEAR-CONFIRMATION, and DCE-CLEAR-CONFIRMATION)

These packets only convey limited information regarding the subscriber ID and availability information. Therefore, a maximum of 128 bits is enough.

SECTION IX

TESTBED SIMULATOR

As indicated in Section VII, to combat fading in mobile channel environments, various error-control techniques, such as trellis code, block interleaving, and packet replication, become necessary. Consequently, it becomes intractable to analyze a system that has a communication channel with memory. Simulation effort becomes crucial for system evaluation. Two testbed simulators were developed in 1988: One is for the request channel in which various types of CAPs can be implemented, the other is for the transmission channel in which different link-connection protocols can be implemented. These two testbeds are discussed in detail in the following subsections. Using the two simulators, one can experimentally study the delay-throughput performance for various protocols in mobile fading environments. The testbeds were designed so that, if the protocol is changed, the structure can be changed easily simply by replacing subsystem black boxes (hardware or software) accordingly.

A. TESTBED FOR REQUEST-CHANNEL SIMULATOR

The purpose of developing this testbed is to simulate the request-channel operation regardless of the type of the CAP used. Figure 9-1 shows a structure of the request-channel testbed. It consists of N superusers (SUs), one collision controller (CC), one physical layer simulator (PLS), and one protocol controller (PTC). Each SU is capable of emulating a number of users, for example, M users. It generates request packets and simulates the initiation of each request on behalf of all M emulated users. Within each SU, a transmission controller (TC) governs the transmission and retransmission of each request. Note that during the simulation, the collision can be recognized by the generation time of each individual packet. Hence, in the testbed, a CC is utilized to predetect any collision before the packets enter the PLS, and to resolve the collision as well. Colliding packets are not sent to the PLS. This structure not only performs the CAP more efficiently but also alleviates the load of the PLS, which is the bottleneck of the testbed. In addition, to implement a different CAP, a different software module can be used in the CC without affecting other subsystems of the testbed. The PLS performs the encoding, interleaving, modulation, fading channel, demodulation, deinterleaving, and decoding. Finally, the PTC is used to detect the packet error and send an acknowledgment to the corresponding SU if a packet is received correctly.

In an operational system, the return channel (for sending acknowledgments) can also be corrupted by channel noise and fading. Hence, another PLS should be inserted for simulating the return channel. Such a configuration has been implemented. However, at the present time, the software interface has not been finished. Therefore, the results with two PLSs for both forward and return links are not included in this report. A supplement to this report will be issued when the software interface is completed.

Figure 9-2 shows the hardware configuration of this testbed simulator. Two SUs are shown in the figure. Each of them is implemented by an IBM XT. The CC, PTC, and PLS reside in three separate IBM ATs (or compatibles) and a Compaq 386. Since multiple communication links are required between the CC and SUs, and between the PTC and SUs, a multiple-port asynchronous board is installed in the CC and PTC. Packets are exchanged between computers at rates of up to 9600 bps. The PLS performs the bit-by-bit simulation on a 20-MHz 68020/68881 coprocessor board installed in the Compaq 386. It runs in the background, allowing foreground processing of graphic displays in real time. Signal-to-noise ratio and the propagation-delay parameters can be changed dynamically to reflect deep fade conditions during simulation. Functions performed by each microcomputer are summarized in Figure 9-2.
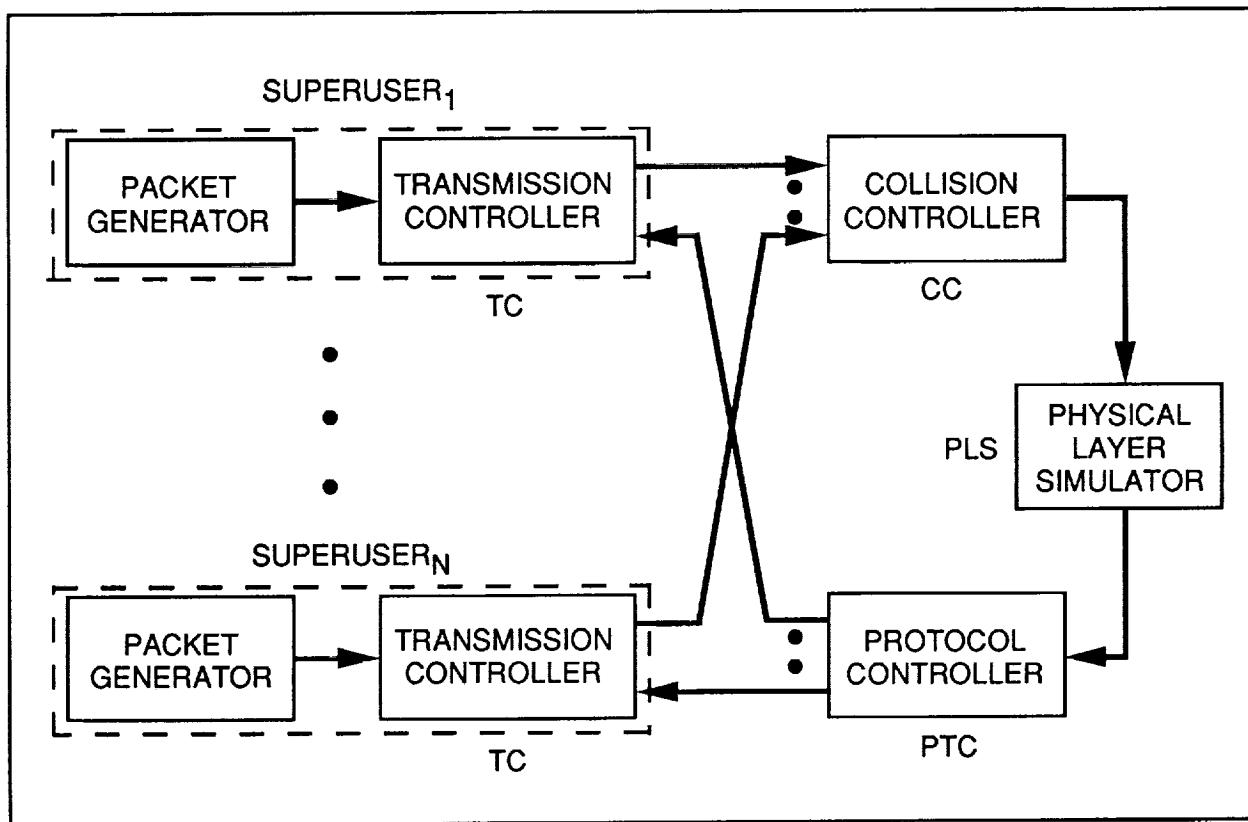
Figure 9-1. Conceptual Structure of the Request-Channel Testbed Simulator

Following the scenario projected for the MSAT-X network, experiments using this request-channel testbed have been conducted to simulate the previously described CAPs: the modified slotted ALOHA protocol and the free-access tree algorithm. The PLS for these tasks includes interleaving (deinterleaving), 16-state, rate 2/3 trellis encoding (decoding) with 8-ary DPSK modulation (demodulation). The data rate is 4800 bps. The request packet is 128 bits long, including the standard Consultative Committee of International Telephone and Telegraph (CCITT) 16-bit CRC sequence for error detection. The propagation delay between the MT and the NMC is 250 ms and the guard time is 16 ms. The processing time for the PLS is 5 ms plus 2 packet times (for interleaving and deinterleaving). For both CAPs, the slot time is the length of the request packet plus the guard time. The time-out period for retransmissions is 12 slots, and the initial backoff delay is uniformly distributed between 1 and 20 slots. The maximum number of retransmissions for each packet is set to 2. In the task of simulating the free-access tree algorithm, two logical channels are implemented.

Figure 9-3 shows the simulation results of the request-channel delay (from the time when the request is made to the time when it is successfully received) for both CAPs when E_b/N_0 = 12 dB. A 10-dB Rician fading parameter and a 20-Hz Doppler frequency are used for the fading channel in the task. A theoretical minimum delay of 8.2 slots (error-free with no collision) is also indicated in the figure for comparison. From Fig. 9-3, one can see that, in the case of modified slotted ALOHA, the average delay increases dramatically when throughput goes beyond 0.2, meaning that the system becomes unstable. However, the average delay for the free-access tree algorithm still remains relatively flat, as long as the throughput is less than 0.25. Comparing these two CAPs, the tree algorithm shows a significant throughput improvement over the modified slotted ALOHA. This result is consistent with the analytical result given in Section V for the case of 2 channels.
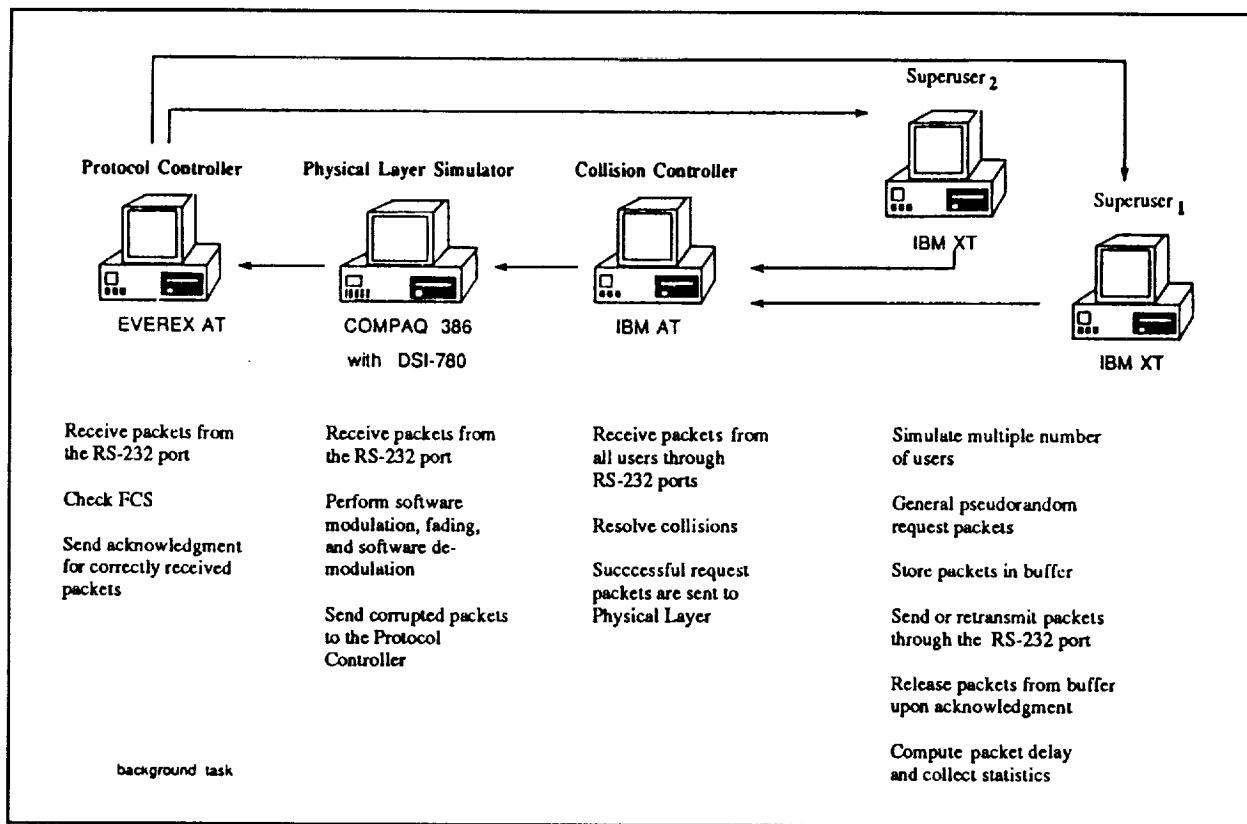
Figure 9-2. Hardware Configuration of the Request-Channel Testbed

B.   TESTBED FOR LINK-PROTOCOL SIMULATOR

The link-protocol simulator consists of two IBM XTs and two IBM AT (or Compaq 386) microcomputers, as shown in Fig. 9-4. Functions performed by each hardware component are summarized in this figure. The IBM XTs generate the real data frames and perform the end-to-end protocol functions. The frames are sent via an asynchronous communication board to, and then are processed by, the IBM ATs. The process includes the fading channel simulation and graphic display. Two IBM ATs are also connected by an asynchronous communication link. After the process, the second IBM AT sends the frames to the other IBM XT via another asynchronous communication board for the protocol processing. The advantage of using the asynchronous link is twofold: First, it is possible to examine the contents of frames on the IBM AT by using a commercially available terminal-emulation program. This facilitates the system debugging. Second, it enables the redirection of data frames to the field-test equipment. Thus, one can test the protocol under operational conditions. An example of this setup is to connect the two ending IBM XTs with a currently existing satellite system through a modem. In this case, one can easily demonstrate the performance of the data-link protocol in a satellite system. It should be pointed out that, by testing the data-link protocol with an operating communication system, the effect of the correlation between two adjacent packet transmissions due to the memory of the channel can be included. The results from field tests can be compared with the simulation results obtained by exercising the testbed.
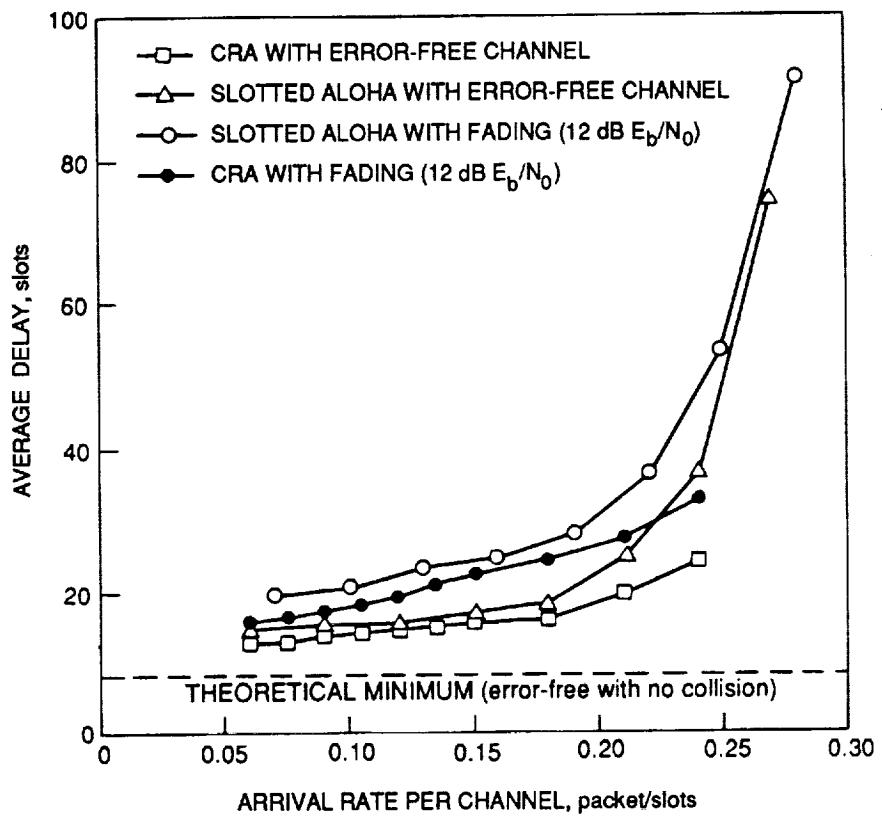
Figure 9-3. Simulation Results of the Request-Channel Testbed

Several software modules are written to connect the hardware. They are

1. Packet-level protocol software
2. Link-level protocol software
3. Driver for asynchronous communication adapter
4. Asynchronous communication-adapter interface software
5. Propagation delay simulation
6. PLS
7. User interface

An example of the packet-level protocol software is the program emulating the MSAT-X connection protocols. The ARQ software incorporates the stop-and-wait, go-back-N, and selective repeat techniques. The driver software program stated above in item 3 includes four modules, namely, (1) frame receiving, (2) frame transmitting, (3) preprocessing frame preparation, and (4) frame-error and flag-error detection. All four modules are written in the 80286 assembly language for the maximum speed. The first two modules involve low-level hardware interface programming for the purpose of hardware interrupt and direct memory access (DMA) handling. The third module is to form the frames into a pattern that can be accessed by the PLS. The fourth module processes the frames after the PLS and detects the frame errors and flag errors. An incorrectly received frame will not be sent to the destination IBM XT; in other words, that frame is lost. The propagation delay is simulated by programming the timer-ticker interrupt of the IBM AT microcomputer. Every frame received by the IBM AT is time-tagged at its arrival and is queued in the transmission buffer. On the other hand,

Destination — IBM XT

Respond to incoming
frame and packet-level
messages

Check FCS

Send acknowledgment
for correctly received
packets

background task

Downlink PC — IBM AT

Receive packets from
uplink PC

Send packets
asynchronously to
destination PC

Perform software
modulation, fading,
and software de-
modulation for
return messages

Strip JPL FLAG and
FCS off the packet

Attach FLAG and FCS

Display real-time graphics

Uplink PC — COMPAQ 386 with DSI-780

Receive data from
origination PC

Strip FLAG and FCS

Attach JPL FLAG and
FCS to packet

Perform software
modulation, fading,
and software de-
modulation

Check FCS

Send data packet
asynchronously to
downlink PC

Display real-time graphics

Origination — IBM XT

Generate frame and
packet-level messages

Collect high-level
statistics

Figure 9-4. Hardware Configuration of the Link-Protocol Testbed

during each timer-ticker interrupt, the transmission buffer is examined, and all frames satisfying the propagation delay requirement are transmitted.

Following the same scenario described in the request-channel testbed, the stop-and-wait, go-back-N and selective repeat ARQ protocols are implemented. The PLS is identical to that in the request-channel testbed. The data rate is 4,800 bps, and the data packet is 256 bits long, including the standard CCITT 16-bit CRC sequence. At the present time, only the performance of the stop-and-wait link-level protocol was examined. (A supplementary document will be prepared after the performance of other protocols is examined.) This simple protocol allows one to calibrate and validate the software modules discussed previously. The link-protocol simulator is first tested in the nonfading additive white Gaussian noise (AWGN) environment with $E_b/N_0 = 12$ dB. Simulation results match closely the theoretical results. Then, a mobile fading environment with a 10-dB Rician fading parameter and 20-Hz Doppler frequency is used in the testbed. Figures 9-5 and 9-6 show, respectively, the throughput and average frame delay of the stop-and-wait protocol.
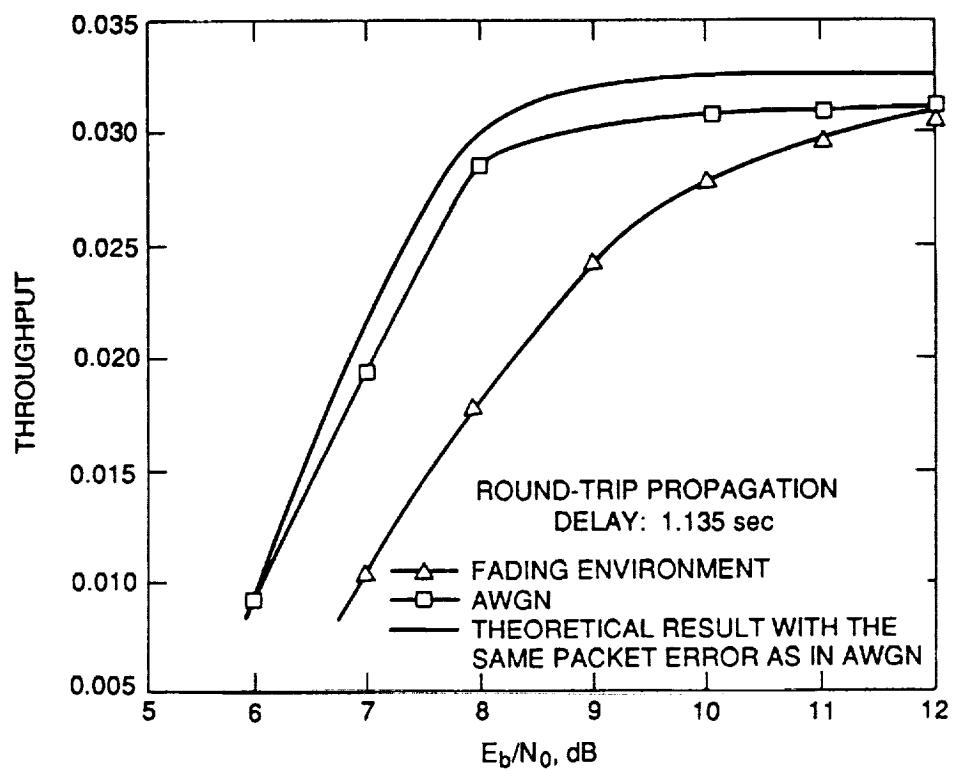
Figure 9-5. Simulation Results of Throughput of the Stop-and-Wait Protocol
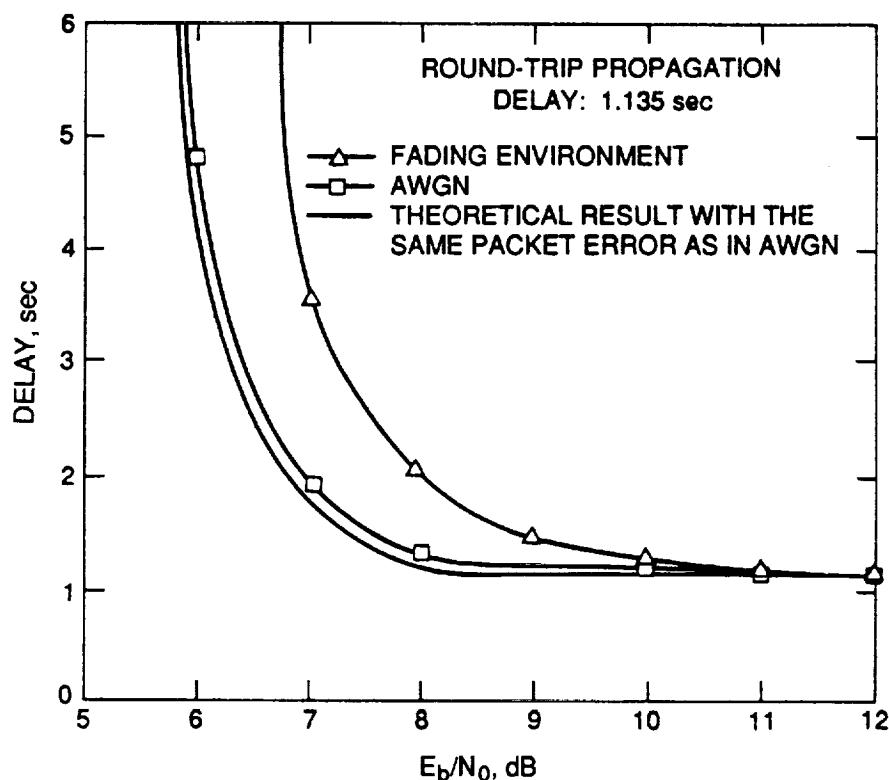
Figure 9-6. Simulation Results of Average Frame Delay of Stop-and-Wait Protocol

SECTION X

A SAMPLE NETWORK CONFIGURATION

This section describes how to obtain the operational characteristics of the MSAT-X network by using the I-AMAP algorithm and a network configuration that may be applicable to the forthcoming MSS.

The example presented here is similar to the filing of the American Mobile Satellite Corporation (AMSC) with the FCC. This first-generation commercial MSS uses an L-band antenna covering the continental United States (CONUS) with 4 spot beams, as depicted in Fig. 10-1. The outer beams will reuse the same frequencies, implying a frequency reuse factor of 1.33.

A.   SPECTRAL EFFICIENCY

To evaluate the system utilization, the network-system spectral efficiency is defined as the number of information bits that the system can convey per second per Hz. Furthermore, the maximum theoretical spectral efficiency Γ_{max} is that obtained strictly from the total number of channels dictated by the modulation and frequency reuse, the data rate per channel, and the total bandwidth, i.e., without taking into consideration the effects of the efficiency of the network protocol used. Hence,

$$\Gamma_{max} = \frac{(bit\ rate\ x\ maximum\ number\ of\ channels)}{total\ bandwidth}$$
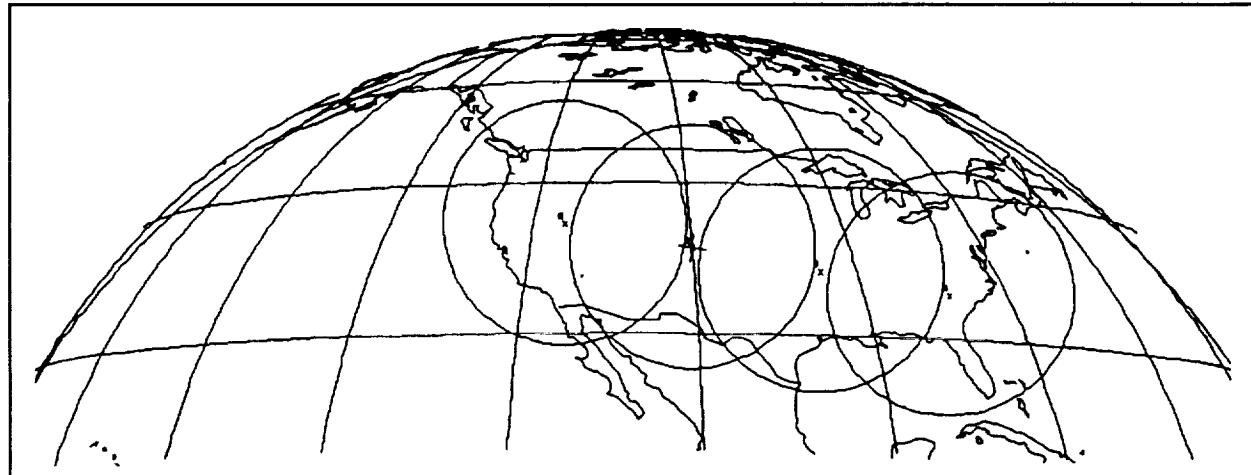

Figure 10-1. Antenna Beam Coverage for First-Generation LMSS

As designated by the 1987 International World Administrative Radio Conference (WARC), the MSS is assigned 7 MHz of uplink and 7 MHz of downlink bandwidth on a primary and coequal primary basis. The uplink (mobile-to-satellite) band(s) are located roughly around 1.65 GHz, while the downlink (satellite-to-mobile) band(s) are located in the vicinity of 1.55 GHz. The results presented herein reflect these regulatory constraints.

Using the FDMA architecture, each 7 MHz is divided into 1,400 5-KHz channels. Assuming that the MTs are uniformly distributed over the CONUS, the 1,400 channels are evenly distributed among 3 beams (the fourth beam at the east end reuses the same channels of the first beam at the west end), resulting in 466 channels per beam, or a total of 1,864 channels for the entire system. Since at least
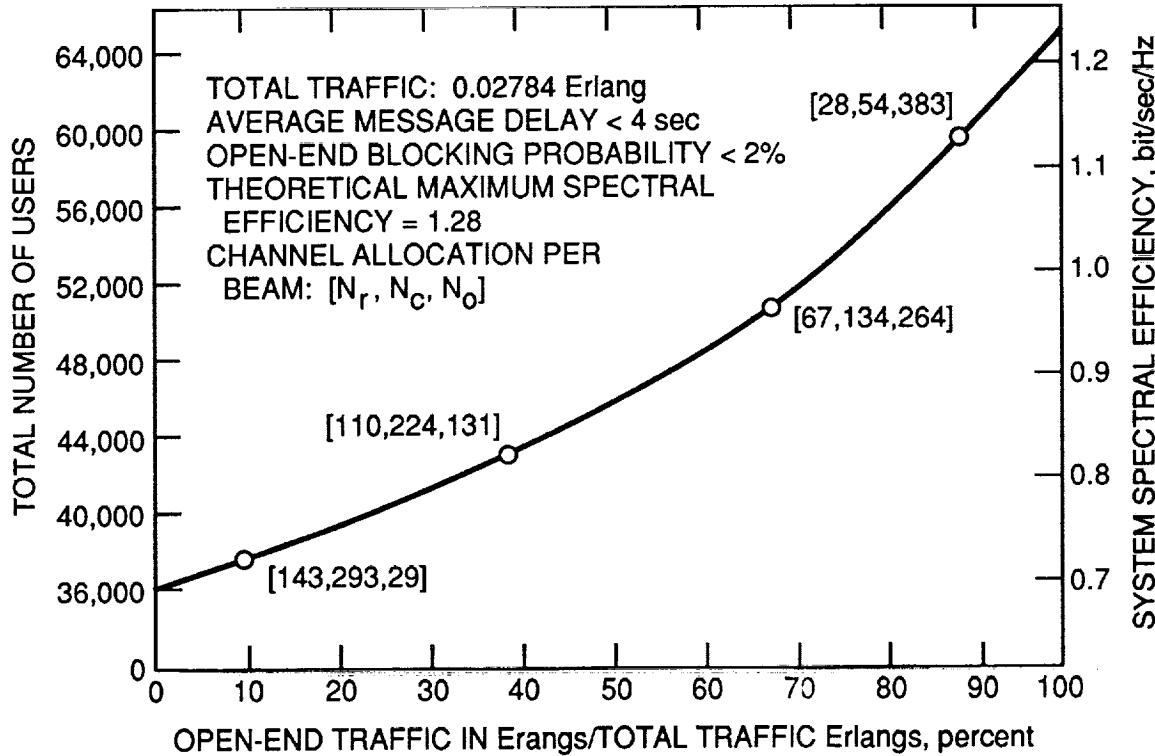
Figure 10-2. System Performance Versus Traffic Mix

one channel is required within each spot beam for new MTs to log on the network, a maximum of 465 channels per beam is considered for making connection requests, closed-end connections, and open-end connections. With 1,860 channels, 4,800 bps through 5 kHz, and 7 MHz total bandwidth, Γ_{max} = 1.28 bps/Hz.

In the MSAT-X, a two-satellite configuration is envisioned to double the capacity of the first-generation system. Two-satellite operation is accomplished by using the discrimination available on the proposed steerable medium-gain vehicle antennas. This discrimination is complemented by polarization isolation to provide a minimum of 20-dB intersatellite isolation. Under this scenario, orbital reuse is accomplished, and the entire 7-MHz bandwidth is used with each satellite. This doubles system capacity, and the theoretical spectral efficiency becomes 2.56 bps/Hz. The network-access protocol remains unchanged within each beam.
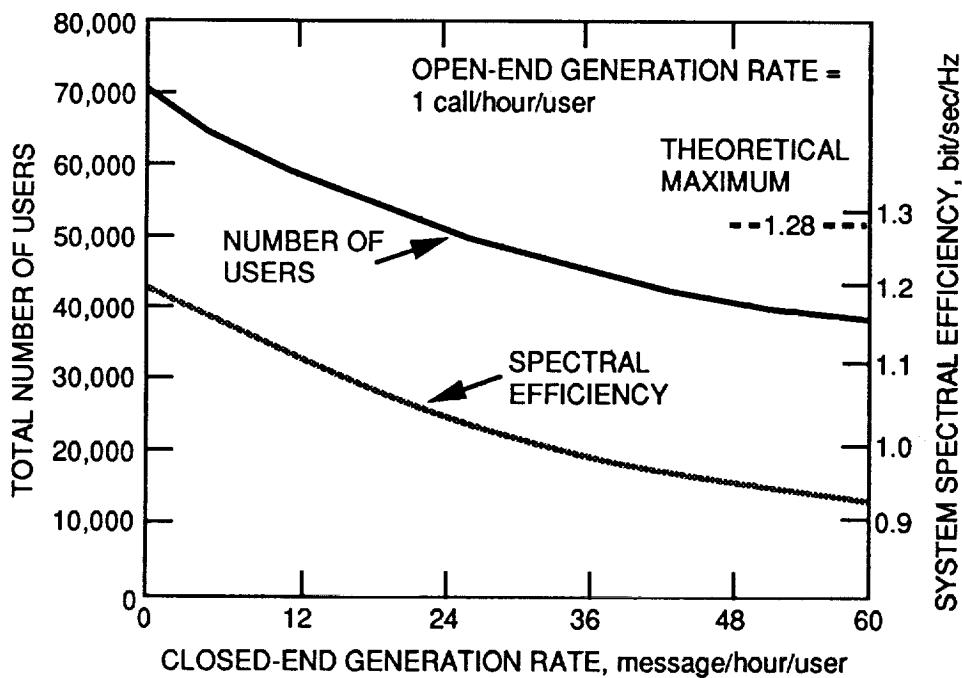
Figure 10-3. System Performance Versus Data Traffic

B. PROJECTED SYSTEM CAPACITY

There are several important parameters that a system designer must be able to obtain, based on the scenario and grade of service considered. These are the channel boundary that the I-AMAP must follow, the number of subscribers that the system can support, and so on. Software has been implemented to calculate these parameters by following the analyses in [5] and [12]. Considering a scenario wherein, on the average, each MT generates twelve 4,096-bit closed-end data messages and one 90-second-long open-end file or telephone call per hour at 4,800 bps, the closed-end traffic becomes 0.00284 Erlang, and the open-end traffic becomes 0.025 Erlang. This reflects a 9-to-1 traffic mix. The Rician fading parameter and Doppler frequency are set to be 10 dB and 100 Hz, respectively. The required BER is 10^{-3}. The modified slotted ALOHA is used for the CAP. Using the recommended packet sizes described previously, one can calculate that the above-described MSS architecture can support up to 58,500 subscribers for the entire four-beam coverage under the constraint of 2-percent blocking probability for open-end connections and 4-second average message delay for closed-end connections. The optimal channel allocation for this traffic mix can also be calculated to be [N_r (request), N_c (closed-end), N_0 (open-end)] = [28, 54, 383]. This shows as the upper point on the curve in Fig. 10-2. The corresponding system spectral efficiency is 1.118 bps/Hz — a remarkable 87 percent of the theoretical maximum. This high percentage of spectral efficiency demonstrates how well the protocol matches the FDMA architecture.
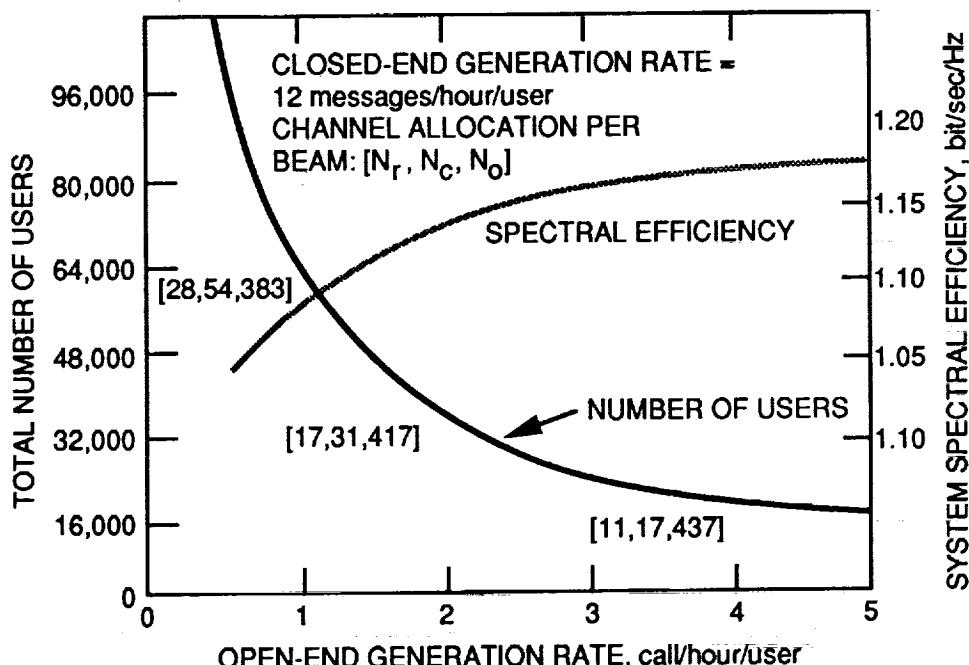
Figure 10-4. System Performance Versus Voice Traffic

Extensive numerical results have been compiled for a wide range of traffic conditions. If one fixes the total traffic at 0.02784 Erlang and varies the traffic mix, the total number of subscribers that the system can support and the system spectral efficiency are shown in Fig. 10-2. Note that, since each closed-end data message takes only about 1/105 of the duration of an open-end connection, the same amount of closed-end traffic (in Erlangs) as open-end traffic requires approximately 105 times the request capacity. This obviously means that, as the closed-end traffic increases, the number of required request channels N_r will increase considerably, i.e., substantially more system overhead will be required. It can be seen in Fig. 10-2 also that the spectral efficiency decreases as closed-end traffic increases. Hence, it can be concluded that the CAP must be improved to handle the increase in request packets for the situation where a higher percentage of closed-end traffic exists. This highlights the necessity for developing the free-access tree algorithm, as described in Section V, that is particularly suited for closed-end dominated traffic. The tree algorithm provides up to 40 percent higher stable throughput per channel than that of the slotted ALOHA.

One can now allow the total traffic to vary, due to the variation in either open-end or closed-end traffic, while the other is held constant. If one fixes the amount of open-end traffic and increases the closed-end traffic, it is obvious that the number of subscribers that the system can support will decrease, since the total amount of traffic generated by each individual subscriber increases. Figure 10-3 shows the variation of the total number of subscribers as a function of the closed-end traffic-generation rate when the open-end traffic-generation rate is fixed at one time per hour per subscriber. Also as seen in Fig. 10-3, increasing the closed-end traffic decreases the system spectral efficiency, since more request channels are required. In contrast, if one fixes the amount of closed-end traffic and increases the open-

end traffic, the system spectral efficiency increases, as seen in Fig. 10-4. This occurs because the system is more efficiently utilized by a more dominant open-end traffic-generation rate. (Yet the number of users now decreases, since each user uses the system much longer.) However, from the user's point of view, the data message is more efficient than the voice message, since data can convey more information per bit.

SECTION XI

FUTURE RESEARCH ACTIVITIES OF THE MSAT-X

Research will continue in both the theoretical and experimental aspects of the MSAT-X network activities, if the budgetary conditions allow. Listed below are some of the possible activities.

A. SELECTION OF THE ERROR-CONTROL TECHNIQUE

As indicated in this report, the examination of go-back-N and selective repeat ARQ techniques using the link-protocol testbed with the inclusion of a separate PLS for the return channel has not been completed. Extensive simulation will be performed using the testbed.

B. INTEGRATED TESTBED SIMULATOR

The testbeds described in this document were not integrated; that is, separate experiments were conducted on each testbed. As planned, a testbed integrating the two simulators will be developed, which can start with the initiation of requests and end with the successful reception of the acknowledgment.

C. INTEGRATED SERVICES DIGITAL NETWORK

The protocols developed under the MSAT-X have many resemblances to the Integrated Service Digital Network (ISDN) developed for the wire-line networks. The combination of the I-AMAP algorithm and the architecture for the wire-line ISDN represents a major effort toward implementing an end-to-end digital pipe between mobile and land or between mobile and mobile subscribers.

SECTION XII

ACKNOWLEDGMENT

SECTION XIII

GLOSSARY OF ACRONYMS

ACK   positive acknowledgment packet

AWGN   additive white Gaussian noise

AMSC   American Mobile Satellite Corporation

ARQ   automatic retransmission request

BS   base station

CAP   channel access protocol

CC   collision controller

CCITT   Consultative Committee of International Telephone and Telegraph

CRA   collision resolution algorithm

CRC   cyclic redundancy check

DAMA   demand assigned multiple access

DCE   data communication equipment

DMA   direct memory access

DPSK   differential phase-shift keying

DTE   data terminal equipment

EOF   end of file

FCC   Federal Communications Commission

FCS   frame check sequence

FDMA   frequency division multiple access

FEC   forward error correction

FET   first exit time

GW   gateway

I-AMAP   integrated-adaptive mobile access protocol

MSS   mobile satellite system

MT   mobile terminal

NAK	negative acknowledgment packet
NMC	network management center
PER	packet error rate
PLS	physical layer simulator
PTC	protocol controller
RETX	retransmission
SCPC	single channel per carrier
SHF	super high frequency
SU	superuser
TC	transmission controller
TCM	trellis-coded modulation

SECTION XIV

REFERENCES

1.   G.H. Knouse and P.A. Castruccio, "The Concept of an Integrated Terrestrial/Land Mobile Satellite System," *IEEE Trans. on Veh. Tech.*, Vol. VT-30, August 1981, pp. 97–101.

2.   R.E. Anderson, R.L. Frey, and J.R. Lewis, "Technical Feasibility of Satellite-Aided Land Mobile Radio," *Proc. Inter. Conf. on Comm.*, June 1982, pp. 7H2.1–7H2.5.

3.   F. Naderi, G.H. Knouse, and W.J. Weber, "NASA's Mobile Satellite Communications Program; Ground and Space Segment Technologies," *Proceedings of the International Astronautics Federation Congress*, Laussane, Switzerland, October, 1984.

4.   W.J. Weber and F. Naderi, "NASA Mobile Satellite Experiment (MSAT-X)," *Proc. National Electronic Conf.*, Vol. 37, Chicago, Illinois, June 1983, pp. 328–330.

5.   V.O.K. Li and T.-Y. Yan, "An Integrated Voice and Data Multiple Access Scheme for a Land Mobile Satellite System," *Proc. of the IEEE*, Special Issue on Satellite Communication Networks, Vol. 72, No. 11, Nov. 1984.

6.   "MSAT-X Networking System Study: Task 1 – I-AMAP Feasibility Assessment, Final Report No. 1," Signatron, Inc., June 8, 1985.

7.   H. Tan and T.-Y. Yan, "A New Free-Access Collision Resolution Algorithm Based Connection Request Procedure for Land Mobile Satellite Systems," *Proceedings of INFOCOM'89*, Ottawa, Canada, April 1989.

8.   H. Tan and H. Wang, "Performance of Multiple Parallel Slotted ALOHA Channels," *Proceedings of INFOCOM'87*, San Francisco, April 1987.

9.   T.-Y. Yan and L.P. Clare, "Performance Analysis of Replication ALOHA for Fading Mobile Communications Channels," *IEEE Trans. on Comm.*, COM-34, Dec. 1986.

10.   OST Bulletin No. 53, "Cellular System Mobile – Land Station Compatibility Specification," Federal Communications Commission, July 1983.

11.   D. Divsalar and M.K. Simon, "Trellis-Coded Modulation for 4800 to 9600 bps Transmission Over a Fading Satellite Channel," JPL Publication 86-8, Pasadena, California, June 1, 1986.

12.   C.C. Wang and T.-Y. Yan, "Performance Analysis of An Optimal File Transfer Protocol for Integrated Mobile Satellite Services," *Proceedings of GLOBECOM'87*, Tokyo, Japan, December, 1987.

13.   K. Dessouky and M. K. Sue, "MSAT-X: A Technical Introduction and Status Report," JPL Publication 88-12, Pasadena, California, April 15, 1988.

1. Report No. 90-19	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle	5. Report Date
Mobile Satellite Experiment (MSAT-X) Network Definition	August 15, 1990
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Charles C. Wang and Tsun-Yee Yan	

9. Performing Organization Name and Address	10. Work Unit No.
JET PROPULSION LABORATORY California Institute of Technology 4800 Oak Grove Drive Pasadena, California 91109	11. Contract or Grant No. NAS7-918
	13. Type of Report and Period Covered External Report JPL Publication

12. Sponsoring Agency Name and Address	
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION Washington, D.C. 20546	14. Sponsoring Agency Code 237-90207-0-3310

15. Supplementary Notes

16. Abstract

This report, No. 164, describes the network architecture development of the Mobile Satellite Experiment (MSAT-X) project for the past few years. The objective of this report is twofold. First, it summarizes the results and findings of the network research activities carried out under the MSAT-X project. Second, it provides a framework upon which the mobile satellite systems (MSSs) operator can design a commercial network. With this goal in mind, a sample network configuration and its capability are included under the projected scenario.

The report is organized into eleven sections. The introduction is Section I, and the MSAT-X network is reviewed in Section II. The Communication Interconnection aspect of the network is discussed in Section III. The MSAT-X network structure is described in Section IV. In this structure, there are two basic protocols: One is the channel access protocol, which is discussed in Section V; the other is the link connection protocol, in Section VI. The error-control techniques used in the MSAT-X project are discussed in Section VII. The packet structure is discussed in Section VIII. A description of two testbeds developed for experimentally simulating the channel access protocol and link control protocol, respectively, is presented in Section IX. A sample network configuration is given in Section X. Finally, Section XI presents some future network activities of the MSAT-X project.

17. Key Words (Selected by Author(s))	18. Distribution Statement
Engineering (General) Communications Electronics and Electrical Engineering	Unclassified, Unlimited

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 53	22. Price